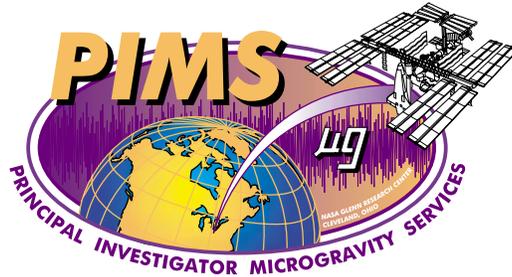


Principal Investigator Microgravity Services



International Space Station System Reference Document PIMS-ISS-100

September 2001
Rev - Baseline



NASA Glenn Research Center
Cleveland, Ohio

Approvals

Prepared by:

Kevin M. McPherson
NASA Glenn Research Center
PIMS Project Manager

Signature:

Review by:

Kenol Jules
NASA Glenn Research Center
PIMS Project Scientist

Signature:

Reviewed by:

Richard DeLombard
NASA Glenn Research Center
Acceleration Measurement Discipline Scientist

Signature:

Reviewed by:

Kenneth Hrovat
ZIN Technologies, Inc.
PIMS Lead Data Analyst

Signature:

Reviewed by:

Nissim Lugasy
ZIN Technologies, Inc.
PIMS Software Engineer

Signature:

Revision History

Revision	Effective Date	Description
Baseline	September, 2001	Baseline Release

Acronyms and Abbreviations

AOS	Acquisition of Signal
BLOB	Binary Large Object
CCSDS	Consultative Committee for Space Data Systems
CM	Center of Mass
COR	Communications Outage Recorder
EHS	Enhanced HOSC System
EXPRESS	EXpedite the PRocessing of Experiments to Space Station
GRC	Glenn Research Center
GSE	Ground Support Equipment
HiRAP	High Resolution Accelerometer Package
HOSC	Huntsville Operations Support Center
ISS	International Space Station
LOS	Loss of Signal
MAMS	Microgravity Acceleration Measurement System
MOIDP	MAMS OSS Interim Data Processor
MMAP	Microgravity Measurement and Analysis Project
MSFC	Marshall Space Flight Center
NASA	National Aeronautics and Space Administration
OARE	Orbital Acceleration Research Experiment
OLMSA	Office of Life and Microgravity Sciences and Applications
OSS	OARE Sensor Subsystem
PAD	PIMS Acceleration Data
PCSA	Principal Component Spectral Analysis
PDSS	Payload Data Services System
PI	Principal Investigator
PIMS	Principal Investigator Microgravity Services
QTH	Quasi-steady Three-Dimensional Histogram
RMS	Root-Mean-Square
RSS	Root-Sum-Square
RTS	Remote Triaxial Sensor System
SAMS-II	Space Acceleration Measurement System-II
SDDS	Science Data Distribution Server
SE	Sensor Enclosure
SSA	Space Station Analysis
TMF	Trimmean Filter
TSC	Telescience Support Center
WWW	World Wide Web

Table of Contents

Approvals.....	i
Revision History	ii
Acronyms and Abbreviations	iii
Table of Contents.....	iv
1 Introduction.....	1
2 Background and Scope	1
3 ISS Components General Description	1
3.1 Data Types	2
3.2 Telemetry Packet Structure.....	3
3.3 Telemetry Packet Routing.....	4
3.4 Data Flow to PIMS Archives.....	5
3.5 TSC Services.....	10
4 PIMS Custom Software Description.....	11
4.1 Packet Grabber.....	11
4.2 Listener	14
4.3 MySQL Databases	16
4.4 Configuration Files	27
4.5 Monitor and Control Application.....	43
4.6 packetWriter.....	44
4.7 PIMS Web Page.....	49
4.8 Database Interface Tool	50
4.9 MAMS OSS Interim Data Processor	51
4.10 Data Recovery and Gap Detection Applications	52
5 Initial Operational Configuration.....	53
6 References.....	54
Appendices	
A Directory Locations for PIMS Software Applications.....	55
B Ancillary Data Parameters.....	56
C SAMS-II and MAMS Data Rates	59
D TMF Description.....	60
Z Open Work.....	62

1 Introduction

The NASA Glenn Research Center (GRC) Principal Investigator Microgravity Services (PIMS) project supports NASA's Microgravity Research Division Principal Investigators (PIs) by providing acceleration data analysis and interpretation for a variety of microgravity carriers including the International Space Station (ISS), the Space Shuttle, the Russian Mir Space Station, parabolic aircraft, sounding rockets, and drop towers. The PIMS project is funded by the NASA Headquarters Office of Life and Microgravity Sciences and Applications (OLMSA) and is part of the NASA Glenn Research Center's Microgravity Measurement and Analysis Project (MMAP), which integrates the analysis and interpretation component of PIMS with the various NASA sponsored acceleration measurement systems. For the ISS, these acceleration measurement systems include the Space Acceleration Measurement System-II (SAMS-II) and the Microgravity Acceleration Measurement System (MAMS).

During ISS operations, the PIMS project is responsible for receiving, processing, displaying, distributing, and archiving the acceleration data from SAMS-II. For MAMS, in addition to the responsibilities for the acceleration data, PIMS is responsible for the text messages and housekeeping data transmitted by MAMS during operations. These tasks are accomplished by utilizing various ISS resources and by utilizing custom software components developed by PIMS. Detailed capabilities of the PIMS custom software components related to PI support are provided in PIMS-ISS-001 [1].

2 Background and Scope

This document is intended as a reference document to describe the overall system employed by the PIMS project to receive, process, and distribute acceleration data from the SAMS-II and MAMS ISS payloads and the Ground Support Equipment (GSE) telemetry packet data required by PIMS to support offline analysis of the MAMS data. While the focus of this document will be on the various components controlled by the PIMS project and the overall data flow through these components, the MAMS and SAMS-II payloads and various aspects of the ISS program will be briefly described in order to explain their interaction with the overall PIMS system. These ISS program items will be discussed in Section 3 while a general description of the PIMS controlled components will be discussed in Section 4.

3 ISS Components General Description

Both SAMS-II and MAMS are payloads in EXpedite the PROcessing of Experiments to Space Station (EXPRESS) racks located in the United States Laboratory module Destiny. Depending on the microgravity requirements of an individual payload, an experiment can be located in either

an Active Rack Isolation System (ARIS) equipped EXPRESS rack or in a non-ARIS equipped EXPRESS rack.

ISS flight 6A marks the beginning of operations for both SAMS-II and MAMS. For flight 6A, two EXPRESS racks are among the complement of racks located in the Destiny module. During increment 2, EXPRESS rack #1 is a “continuously” powered, non-ARIS equipped rack and EXPRESS rack #2 is an ARIS equipped rack. ARIS equipped EXPRESS racks have a SAMS-II RTS embedded within the rack. During initial operations, MAMS and two RTS drawers are located in EXPRESS rack #1. The SAMS-II Interim Control Unit (ICU) drawer and a third RTS (embedded RTS) are located in EXPRESS rack #2.

3.1 Data Types

3.1.1 MAMS

MAMS is located in a single locker location and consists of two accelerometer systems located within this locker volume. The Orbital Acceleration Research Experiment (OARE) Sensor Subsystem (OSS) is designed to measure the quasi-steady accelerations of the ISS as it moves through its low Earth orbit. The High Resolution Accelerometer Package (HiRAP) is designed to measure the vibratory/transient acceleration environment of the ISS at the MAMS location. Reference [2] provides further explanation of these two acceleration regimes.

3.1.2 SAMS-II

The SAMS-II consists of two primary components, distributed as required throughout the ISS to measure the vibratory/transient acceleration environment. The Interim Control Unit (ICU) is a Portable Computer System (PCS) laptop configured to support the SAMS-II system. The ICU serves as a collection point for all SAMS-II acceleration data and all SAMS-II housekeeping data. Any commands intended for the SAMS-II and its components are routed through the ICU. The second component is called the Remote Triaxial Sensor Subsystem (RTS) and consists of an Electronics Enclosure (EE) and two Sensor Enclosures (SE). The RTS is located throughout the ISS to provide a distributed, localized measurement of the microgravity acceleration environment. The SE contains the actual accelerometers and transmits its acceleration data to the EE and subsequently to the ICU for downlink to the ground.

3.1.3 GSE Packet Data

The GSE packet data used by PIMS are generated using the Enhanced Huntsville Operations Support Center (HOSC) System (EHS). A GSE packet is assembled on the ground by a user to create a custom telemetry packet composed of any readings defined in the EHS telemetry database. For PIMS, the contents of these GSE packets contain ISS center of mass (CM) data and ISS body rates and body angles data necessary to mathematically map the MAMS OSS data from its measured location to the experiment location of interest.

3.2 Telemetry Packet Structure

3.2.1 CCSDS Telemetry Packet Structure

Both payloads communicate with the EXPRESS rack, and ultimately with the ISS, through the EXPRESS Rack Interface Controller (RIC). Of particular concern for PIMS is the transmission of telemetry packets from MAMS and SAMS-II. In order to properly communicate with the RIC, each payload must prepend an EXPRESS primary and secondary header to each telemetry data packet. The contents of these headers instruct the RIC on the type of telemetry packet, the source of the telemetry packet (source function code), and the destination of the telemetry packet (destination function code). The telemetry packet structure, including the lengths of the primary and secondary EXPRESS telemetry packet headers, are shown in Table 1.

EXPRESS Header	Secondary EXPRESS Header	Acceleration Data
8 bytes	8 bytes	

Table 1 - MAMS and SAMS-II Telemetry Packet Structure with EXPRESS Headers

After receiving a telemetry packet, the RIC determines the source function code and prepends the primary and secondary CCSDS headers to the as-received telemetry packet. There are two important parameters contained in the CCSDS headers the RIC prepends. The first parameter is the Application Process Identifier (APID) which is a unique number assigned to each payload and is used on the ground to distinguish one payload's data from another. The RIC uses the source function code in the EXPRESS header of the received telemetry packet to determine the APID value to include in the CCSDS headers. The second important parameter is the Packet Sequence Counter (PSC). This counter is unique for each APID and increments each time the RIC receives a telemetry packet from the payload identified by the source function code. The telemetry packet structure following the addition of the CCSDS headers is shown in Table 2.

CCSDS Primary Header	CCSDS Secondary Header	EXPRESS Header	Secondary EXPRESS Header	Acceleration Data
6 bytes	10 bytes	8 bytes	8 bytes	

Table 2 - MAMS and SAMS-II Telemetry Packet Structure with CCSDS Headers

On-board the ISS, the telemetry packet is sent from the EXPRESS rack to the Automated Payload Switch (APS). The output of the APS is routed to the Communications Outage Recorder (COR) and to the High Rate Frame Multiplexer (HRFM) for downlink. On the ground, the telemetry packet is ultimately received at the MSFC POIC. The Enhanced HOSC System (EHS) within the POIC prepends an EHS primary and secondary header to each telemetry packet prior to

forwarding the telemetry packet to the payloads via the Payload Data Services System (PDSS). The telemetry packet structure following the addition of the EHS headers is shown in Table 3. The header portion of a CCSDS telemetry packet consists of a total of 60 bytes.

EHS Primary Header	EHS Secondary Header	CCSDS Primary Header	CCSDS Secondary Header	EXPRESS Header	Secondary EXPRESS Header	Acceleration Data
16 bytes	12 bytes	6 bytes	10 bytes	8 bytes	8 bytes	

Table 3 - MAMS and SAMS-II Telemetry Packet Structure with EHS Headers

3.2.2 GSE Telemetry Packet Structure

The format of the GSE packets is different than the format of the MAMS and SAMS-II packets. Since the GSE packet is generated on ground in the EHS, there are no EXPRESS headers. The APID information for the GSE packet is contained within the EHS secondary header and is repeated within the CCSDS primary header. The telemetry packet structure of GSE packets data is shown in Table 4. The header portion of a GSE telemetry packet consists of a total of 39 bytes.

EHS Primary Header	EHS Secondary Header	GSE Primary Header	GSE Secondary Header	Acceleration Data
16 bytes	16 bytes	6 bytes	1 bytes	

Table 4 - GSE Telemetry Packet Structure

3.3 Telemetry Packet Routing

The SAMS-II, MAMS, and GSE telemetry packets are routed via User Datagram Protocol (UDP) to an IP address at the Telescience Support Center (TSC) located at the NASA Glenn Research Center. For a given payload's CCSDS packets, the destination IP address or multicast IP address for the telemetry packets is determined by the APID value contained in the CCSDS header of the telemetry packet. This APID value/destination IP address pair is stored in the PDSS. The PDSS and the TSC data systems define the port number for a given payload data packet based on the data mode. There are four data mode possibilities: real time, COR dump, playback 3 (Line Outage Recorder (LOR) of real time), and playback 5 (LOR of COR dump). Therefore, for a given CCSDS packet, the proper routing of the data packet requires that the correct IP address and port number definitions are known in the PIMS equipment processing CCSDS data packets.

For the GSE telemetry packets, the user initiates the flow of the data and defines the destination IP address and port number for the data at the TSC.

3.4 Data Flow to PIMS Archives

With a brief explanation of the data types and the format of the individual data packets complete, it is important to explain the overall flow of data packets through the PIMS system and ultimately into the PIMS acceleration data archives. A fundamental charter of the PIMS project is to maintain archives of the acceleration data obtained on-board the ISS. This introductory data flow is intended to provide a basic data flow framework from receipt of the data by PIMS to the archival of the data. This basic data flow framework can be referenced during the more detailed discussions of the individual PIMS software components.

3.4.1 Introduction to Data Archive Format

A brief discussion of the archive format is warranted at this time. Regardless of the acceleration data type, each acceleration data archive will actually consist of two files: in general, the first file (binary format) contains the acceleration data itself in the form of time, x-acceleration, y-acceleration, and z-acceleration and the second file (XML format) is referred to as the header file and contains ancillary data that describes the circumstances under which the data were obtained. The MAMS raw acceleration data contains two additional columns in addition to the acceleration data. The additional columns in the MAMS raw acceleration data files are the OSS base temperature and the OSS status bytes necessary for determining bias activity from the raw data stream. For GSE packet data, the first file will contain time plus the parameters selected by PIMS for archival. The current set of ancillary data parameters included in each archive file are:

- T-0
- Sampling Rate
- Cutoff Frequency
- Head ID
- Gain
- Data Quality Measure
- Bias Coefficients
- Scale Factor Data
- Location
- Orientation
- Coordinate System
- Station Configuration

3.4.1.1 Directory Structure for Data Archives

The directory structure for the data archives is based on the time the data were obtained and processed. Figure 1 illustrates the various levels of the directory structure, starting with the year, month, and day designations. Each individual sensor type as described in section 3.4.3 will have its own branch from the basic time based structure.

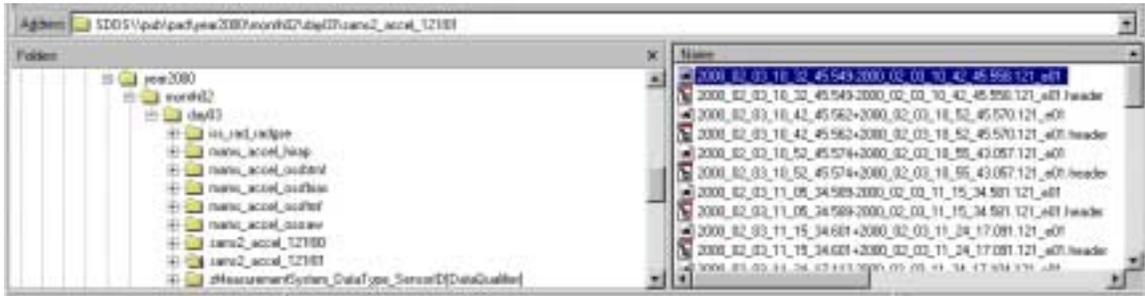
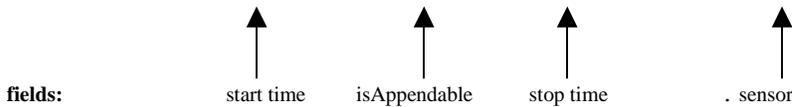


Figure 1 - PAD File Directory Structure

3.4.1.2 Filename Description

The filename convention for the data archives is described below. The XML header file consists of the same file name convention with a “.header” appended to the end of the base filename.

filename: YYYY_MM_DD_hh_mm_ss.sss±YYYY_MM_DD_hh_mm_ss.sss.<SensorID>



The fields are defined as follows:

- start time:** GMT for timestamp of first record in the data file
- isAppendable:** “+” if this data file is appendable to previous; otherwise “-”
- stop time:** GMT for timestamp of last record in the data file
- sensor:** sensor ID

The naming convention for the directory for each of the data types available are described in the Table 5. These data types will be explained in detail in section 4.3.

Data Type	Directory Name
MAMS OSS raw data	mams_accel_ossraw
MAMS OSS best available TMF data	mams_accel_ossbtmf
MAMS OSS final TMF data	mams_accel_ossftmf
MAMS GSE data	iss_rad_radgse
MAMS OSS final bias data	mams_accel_ossfbias
MAMS HiRAP data	mams_accel_hirap
SAMS data	sams2_accel_121xxx, where xxx=serial number of a given SAMS-II sensor

Table 5 - Data Type and Sensor ID

Data gaps and changes in the ancillary data contained in the header file trigger writing a new file, the first of which will have **isAppendable** set to “-“. With this convention, a series of files that constitute a contiguous slice of data will have **isAppendable** set to “-“ in the first file of the series and a “+” in the remaining files in the series. The changes in ancillary data that cause the creation of a new data file are located in Appendix B.

3.4.2 Generic Data Flow Discussion

Before preceding to specific data flow discussions relative to the various data types and their archives, it is important to provide a generic data flow discussion. In general, a data packet is received by a PIMS computer and is inserted into a real-time database appropriate for that data type (see Section 4.1). These real-time database types are discussed in detail in Section 4.3.1. The PIMS real-time display software extracts these data packets from the database (see Section 4.2) and generates the appropriate acceleration data plot.

With the real-time plots generated, the PIMS software must wait approximately 24 hours before generating the archive data files. The waiting period is necessary because of AOS/LOS constraints on the data packet flow. For the ISS, a reasonable estimate places the PIMS system in an AOS period approximately fifty percent of the time. The archive data files must, of course, contain all of the acceleration data generated by the SAMS-II or MAMS instrument over a given period of time.

The ISS COR will record all ISS data during LOS intervals. Periodically, the COR will dump its data to the ground. The MSFC PDSS will distribute this LOS data to the individual payloads. When the LOS data are received by the PIMS computers, the data are automatically inserted into the database in a time ordered fashion. When PIMS has waited the 24 hour interval, PIMS declares that the database has received all the data it was supposed to receive for a given time interval. At this point, the database can be used to begin the process of generating a complete archive data file.

3.4.3 Archive Data Types

For some data types, specialized processing must be performed on the data prior to generating the archive data file. This specialized processing results in the generation of interim databases as described in Section 4.3.2. For the remaining data types, the generation of the archive data file can be accomplished directly from the real-time database. Table 6 below lists the archive data types and the database type used for generating the archive data files.

Archive Data Type	Database Type	Units
SAMS-II Acceleration Data	Real-time	g's
MAMS HiRAP Acceleration Data	Real-time	g's
Raw MAMS OSS Acceleration Data [#]	Real-time	g's
MAMS OSS Bias Data	Interim	g's
Filtered, MAMS OSS Acceleration Data, No Bias ^{%1}	Interim	g's
Final Filtered MAMS OSS Acceleration Data ^{%2}	Interim	g's
Best Available Filtered MAMS OSS Acceleration Data ^{%3}	Interim	g's
GSE Packet Data	Interim	Various

- No TMF or bias compensation applied

%1 - TMF applied and no bias compensation with bias data for on/off cycle, NO archive generated for this data type

%2 - TMF applied and bias compensated with all bias data for on/off cycle

%3 - TMF applied and bias compensated with best available bias data

Table 6 - PIMS Archive Data Types

3.4.4 Real-Time Database versus Interim Database

The purpose of this section is to describe the specialized processing required to generate the required interim database files and to introduce the artificial packet concept utilized in the final processing step before archival of interim data types. Per Table 6, the generation of the archive data files for the SAMS-II acceleration data, the HiRAP acceleration data, and the raw OSS acceleration data is accomplished directly from the real-time database for those data types; no additional processing is required.

3.4.4.1 Artificial Packet Database Format

Prior to being written to the PIMS archives, interim data are written to their individual interim database in the artificial packet format described below. The format of these interim databases consists of a time column and an artificial packet column. The data contained in the artificial packet column are generated by the MAMS OSS Interim Data Processor (MOIDP). The intent of the artificial packet format is to present a standard input to the packetWriter software (see Section 4.6) regardless of the data (SAMS-II, MAMS, GSE packets, etc.) packetWriter is processing.

All samples in the packet must be consecutive so that their time can be determined knowing only the time of the first sample, the sample rate, and the position of the sample within the packet. Non-consecutive samples require separate packets.

Start	Length (bytes)	Description	Notes
0	2	Artificial packet indicator	In hex: FACE
2	1	Artificial packet subtype [#]	0, 1, ...; for MAMS-TMF-best available, ...
3	1	Device id	0, 1, ... in case there are multiple devices of the same subtype (normally 0)
4	8	Start time of first sample (double)	“Unix time” + fractional second
12	4	Sample rate (float)	In Hz, reciprocal of time between samples
16	4	Number of samples in this packet (int)	At least 1
20	4	Measurements per sample (int)	Normally 3: for x, y, z acceleration
24	4	Offset to start of data (int)	Normally 28
28	4...	Data (floats until end of packet)	At least 1

Artificial packet subtype definitions:

0x01 = Best Available TMF, 0x02 = Final TMF, 0x03 = Final Bias, 0x04 = GSE packet data, 0x05 = SAMS-FF data

Table 7 - Artificial Packet Description

3.4.4.2 Interim Databases

The processing to generate the interim database for the GSE packet data consists of a simple averaging of the telemetry contained in the GSE packet. GSE packets are transmitted at a rate of one packet per second. Since the MAMS OSS data is transmitted at a rate of one packet every sixteen seconds, the GSE packet data will be written into the interim database as the simple average of the previous sixteen GSE packets. This averaging is intended to filter out any “bad” telemetry that might be received for a given parameter in the GSE packet. The averaged data in the interim database will be used to generate the GSE packet data archive files. The contents of the interim databases required for GSE packets are described in section 4.3.2.1.

The interim database files required for the filtered, bias compensated MAMS OSS data result from the need to bias compensate the MAMS OSS data before it can be utilized. The bias data are embedded in the raw MAMS OSS data stream. In order to most accurately bias compensate the MAMS OSS data, the complete bias set for a given on/off cycle must be available prior to bias compensating any MAMS OSS data. The data set generated using all the available bias data from a given on/off cycle is referred to as the final filtered MAMS OSS acceleration data. However, in order to service data requests from PI’s received in the middle of an on/off cycle, a filtered MAMS OSS “best available” data set will be generated as a temporary set of files. This “best available” data set is generated using all the bias data available up to the current time in the current on/off cycle. For both MAMS OSS interim databases, the bias data is located in the bias

databases as described in Section 4.3.3.4. Stable MAMS bias values may result in using the best available TMF data as the final TMF filter data, thereby simplifying the MAMS operations.

To accommodate the need for both of these data sets, three interim databases are generated to allow for the creation of both the final filtered MAMS OSS acceleration data and the “best available” MAMS OSS acceleration data. The interim database described in Section 4.3.2.2 contains MAMS OSS data that has been trimmean filtered, but has not had any bias compensation performed. This interim database is utilized to generate both the final filtered MAMS OSS acceleration data and the “best available” MAMS OSS acceleration data.

The interim database described in Section 4.3.2.4 contains MAMS OSS data that has been trimmean filtered and bias compensated. The bias compensation performed for the “best available” data set described in Section 4.3.2.4 utilizes the bias data available up to the current time in the current on/off cycle.

A MAMS on/off cycle could last in length from days to weeks. When the on/off cycle is complete, the entire on/off cycle data set contained in the interim database described in Section 4.3.2.2 is bias compensated enmasse using the complete bias data set stored in the bias database (see Section 4.3.3.4.3). The resulting data set is the interim database file called the “Final Filtered MAMS OSS Acceleration Data” and is described in section 4.3.2.3. This database is used to generate the final filtered MAMS OSS acceleration data archive.

3.5 TSC Services

In addition to providing the network interface for the telemetry packets, the TSC provides two additional functions for PIMS. The mass storage system records all telemetry packets received by the TSC mission network. This includes all SAMS-II, MAMS, and GSE packet telemetry as well as the telemetry packets from other payloads operating at the TSC. In the event of a problem with the PIMS custom software responsible for receiving SAMS-II, MAMS, or GSE data, the mass storage system can be used to request a playback of any lost telemetry packets.

The TSC administers the Science Data Distribution Server (SDDS). This system is used by PIMS as the archive system for all processed PIMS data, including SAMS-II, MAMS, and GSE packet data. The SDDS is the location for the PIMS web server that serves as the main interface between PIMS and PI’s interested in acceleration data.

4 PIMS Custom Software Description

4.1 Packet Grabber

The packetGrabber software serves as the interface between UDP network traffic and the PIMS databases described in Section 4.3. PacketGrabber is a program written in Linux gcc c/c++ and is run as a daemon. It is controlled by a configuration file located in the /etc directory which has a default name of 'packetGrabber.conf'. There is usually one line in packetGrabber.conf for each combination of port and database table to be used. Each line has at least 6 columns of information, and optionally may have more columns if pairs of offset/byte-sequence columns are specified. An example configuration file used during MAMS testing at KSC is shown in Table 8 and a description of the various packetGrabber.conf columns follows.

Port #	Broadcast or Multicast	Database Table Name	Replace or Append	Bytes in Header	Timestamp Type	Offset/Byte Pairs
4956	broadcast	textm	append	60	MAMS	-8 0002
4956	broadcast	hirap	replace	60	MAMS	-8 0001
4956	broadcast	oss	replace	60	MAMS	-8 0003
4956	broadcast	housek	replace	60	MAMS	-8 0004
4956	broadcast	soos	replace	60	MAMS	-8 0005

Table 8 - PacketGrabber.conf Example File

4.1.1 Column 1 – Port Number

The first column is the port number to watch for UDP packets. Multiple lines within a given configuration file can use the same port if multiple types of data are sent to the same port (see Table 8).

4.1.2 Column 2 – UDP Protocol

The second column can have only one of two values, “broadcast” or “multicast”. If broadcast distribution is used, this column provides the broadcast IP address of the appropriate PIMS subnet. If a multicast protocol is employed, this column will be used to hold the IP address of the multicast group for this data type.

4.1.3 Column 3 – SQL Table Name

The third column is the name of the database table in which the captured packets will be stored. Note that SQL table names are not allowed to contain the “-” (minus) character.

4.1.4 Column 4 – Primary Key Setting

The fourth column can have only one of three values, “replace”, “append”, or “ignore”. This column determines how data is written to the database. The “ignore” option prevents data from being written to the database at all; the packets are simply dropped. This is useful if a “catch all”

rule has been added to capture any unknown packets that arrive, but there are certain types of packets that will not be kept. The “ignore” rule will match incoming packets and drop them to prevent them from being written to the “catch all” database. Note that the packet matching rules are processed in the order that they appear in the packetGrabber configuration file. For the ignore rule to work, it must appear before any other rule that might match the packet.

The difference between “replace” and “append” determines whether data with the same timestamp will be overwritten or appended in the database. Note that this is accomplished by creating the database table with a primary key enabled or disabled. This column will have no affect if the database table already exists since PacketGrabber only creates the table if it does not already exist. The only database table currently requiring the “append” option is MAMS text messages. The “replace” option should be used in the majority of the cases.

4.1.5 Column 5 – Number of Header Bytes

The fifth column is the number of header bytes in the packet. Bytes with a zero-based index less than this number will be stored in the “header” column of the database, and bytes with a zero-based index greater than or equal to this number will be stored in the “packet” column of the database (the “payload” portion of the packet).

4.1.6 Column 6 – Time Format

The sixth column is the time format of the packet, which may be one of “SAMS-II”, “MAMS”, “TISS”, “EHS”, or “time-now”. Depending on the packet type, the time information may be located in either the payload portion of the packet or the header portion of the packet. For SAMS-II and MAMS, the time information is located within the payload portion of the packet. For GSE packets, the time information is extracted from the data portion of the data packet in the location of the first GSE packet reading location.

In either case, the packetGrabber code has a hard-coded offset that is used to locate the time information within the appropriate portion of the data packet (i.e. header or payload). These offsets are provided in Table 9. Used in conjunction with the number of header bytes specified in the packetGrabber.conf file (number of header bytes + hard-coded offset) and the time information location in the overall packet (payload portion versus header portion), the packetGrabber software determines the location of the time information within the overall packet.

Packet Type	Time Information Location	Hard-Coded Offset	Overall Offset for Time Information
SAMS-II	Payload Portion	36	Number of Header Bytes (60) + 36
MAMS	Payload Portion	0	Number of Header Bytes (60) + 0
EHS	Header Portion	4	4
TISS	Payload Portion	9	Number of Header Bytes (39) + 9

Table 9 - packetGrabber Hard-Coded Offsets

4.1.7 Optional Offset/Byte Pair Columns

Optional additional columns always come in pairs consisting of a zero-based integer offset into either the header portion or the payload portion of the packet that specifies where to look for matching bytes, and a hexadecimal (2 characters per byte) representation of the bytes to be matched. The offset may be a negative number to count backwards into the header port of the packet. The hexadecimal byte sequence may be of any length. All bytes in the sequence must be matched at the specified offset for the packet to be placed into the database. If there are no offset/byte-sequence pairs specified for a line, then all packets arriving on that port will be accepted into the database.

Table 10 shows the minimum set of offset/byte pairs to be utilized for each packet type that the packetGrabber is required to ingest.

PacketType	RequiredOffset/Byte Pairs #1	RequiredOffset/Byte Pairs #2
SAMS-II	0 DABE	24 3132312D663032 [#]
MAMS	-8 0003 [%]	-32 1B87
GSE	-36 06	-17 199E

- Hex representation of sensor ID, in this case 121-f02 % - Hex representation for MAMS message type, in this case OSS data

Table 10 - Required Offset/Byte Pairs

Table 11 shows a typical packetGrabber configuration file containing the information required to allow packetGrabber to receive and process data packets transmitted using the four available data modes described in Section 3.3. This example is for MAMS OSS data only.

Port #	Broadcast or Multicast	Database Table Name	Replace or Append	Bytes in Header	Timestamp Type	Offset/Byte Pairs
6040	239.1.4.254	oss	replace	60	MAMS	-8 0003
6140	239.1.4.254	oss	replace	60	MAMS	-8 0003
6240	239.1.4.254	oss	replace	60	MAMS	-8 0003
6340	239.1.4.254	oss	replace	60	MAMS	-8 0003

Table 11 - Example packetGrabber.conf File with Data Modes Included

4.2 Listener

The Listener is a program that runs in the background (a Unix *daemon*) watching for MySQL database requests from the network. The listener software serves as the interface between the PIMS databases described in Section 4.3 and the Labview real-time applications and the Matlab offline applications. When Listener gets a request for data, it passes it through to the database and returns any results to the requester. Its purpose is to provide simple remote access to display and analysis programs for the telemetry data collected from the International Space Station for the PIMS project.

PIMS telemetry data is captured on distributed computers and stored in MySQL databases (see Section 4.3). Software for real time displays or offline analysis need to be able to retrieve that data from both local and remote computers. The display and analysis software is usually written in the Labview or Matlab programming languages. Since Labview and Matlab (as well as Python and C, the other languages PIMS software uses) have simple interfaces to network sockets, the Listener program was designed with that interface. A corresponding “talker” Labview virtual instrument (*talker.vi*) was written for the Labview side of the interface. The talker VI is embedded in other Labview code to handle the interaction with the database.

Since Listener is intended to run in the background without user interaction, it does not have much of a user interface. The Listener program itself has no configurable parameters, so it does not require a configuration file or command line parameters when it starts.

4.2.1 Listener Network Protocol and Theory of Operation

Listener sets up a TCP/IP socket on port 4960 and waits for requests to be sent to that port. Listener only serves requests to databases on the same the computer where Listener is running. A client program that wants to retrieve data from the database (such as *talker.vi*), must open a TCP/IP connection to port 4960 on the computer where Listener is running. After a connection is established, the client program sends several parameters to Listener that describe the SQL command for the database to execute. Depending on the outcome of command execution, Listener will send back status, data, or an error message to the client.

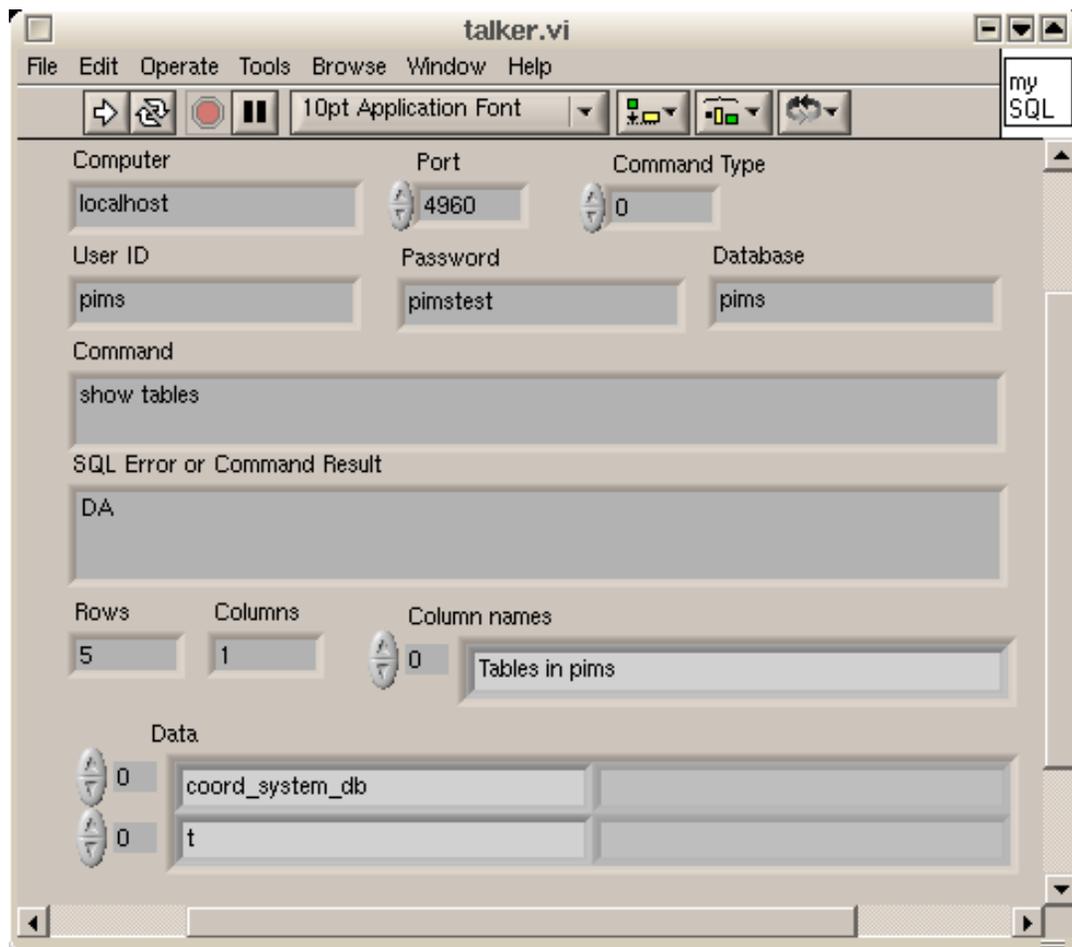
Listener opens a socket on port 4960, and waits for network traffic to appear on the socket. When a connection is made, the Listener process forks so that the database request is handled in a separate process. The original Listener process goes back to watching port 4960 for new connections.

4.2.2 Talker.vi

Even though Listener has no user interface, it does have several protocol parameters that need to be described. All of the Listener protocol items appear in the user interface for talker.vi, so a picture of talker.vi in operation can be used to illustrate the inputs and outputs of Listener.

4.2.2.1 Computer Name and Port Number

The first two fields in the talker.vi diagram are the computer name and port number required to establish a TCP/IP connection.



4.2.2.2 Database Command Parameters

These first four parameters that follow are all sent as ASCII text (command type, user ID, password, and database name). The final two parameters are sent as binary data (SQL command and SQL command length).

Command type is an integer that indicates how the command should be handled by Listener. This was originally included when Listener was envisioned to execute shell commands and well as SQL commands. For SQL commands, the value of command type should always be zero.

The next two parameters are the user ID and password to use when logging into the MySQL database. They are placeholders in the listener code and they can be left empty.

The fourth parameter is the name of the database that is the destination for the command. This field must be present and contain a valid database. The database named “pims” is normally used for PIMS telemetry data.

The final fifth and sixth parameters are the actual SQL command to be sent to the database specified by the fourth parameter. These parameter may include binary data such as SQL “BLOB” (binary large object) fields and are preceded by the command length (number of bytes in the command). The command length is a 32 bit little-endian signed integer. The command itself is just a series of bytes containing the SQL command. Most of the commands sent through the talker.vi are “non-BLOB” related SQL commands are presented as normal ASCII text. The command length does not appear in talker.vi because it is calculated automatically from the command itself.

4.3 MySQL Databases

Various databases are utilized to support PIMS ISS operations. Several of these databases are utilized in the real-time systems that communicate with the packetGrabber and Listener programs. The remaining databases are employed to input and store information that will be included in the PIMS acceleration data archives. These archives and their contents were discussed in Section 3.4.1. Included in the description of each database is a text description, a sample record of data, the source of the data contained in this database, and the input and output mechanisms that control how data arrives in the database and how it is removed from the database.

4.3.1 Real-Time Databases

The databases described in this section contain the UDP packet data obtained from the ISS through the MSFC and TSC data network. The data contained in these databases are the as-received packets obtained by the packetGrabber software. The packetGrabber, through its configuration file, has the ability to separate the UDP data stream by data type into the various table types that are subsequently described in this section.

These databases are the source of data packets for the real-time acceleration data displays. Using the Listener software to extract time and payload data from the database, the real-time software extracts the acceleration data from these packets and creates the appropriate acceleration data

displays. The databases are used by the packetWriter (see Section 4.6) software to generate the acceleration data portion of the PIMS archives.

4.3.1.1 SAMS-II Packet Data Database

A record in the SAMS-II acceleration data database has four columns of data extracted from the original UDP data packet received by the packetGrabber [3]. Since there exist multiple SAMS-II sensors on the ISS at any given time, a database of this format will exist for each SAMS-II sensor on the ISS.

Table Format

Column 1: Time Extracted from the data packet

Column 2: Payload portion of data packet (SAMS-II data)

Column 3: Type of data packet (SAMS-II, MAMS, or GSE data)

Column 4: Header portion of the data packet

The data source for the SAMS-II packet database is the ISS data stream. Data is placed into this database by the packetGrabber and is ultimately removed from the database by the packetWriter.

4.3.1.2 MAMS Database

A record in a MAMS message type database has four columns of data extracted from the original UDP data packet received by the packetGrabber. Since MAMS generates multiple MAMS message types on the ISS at any given time, a database of this format will exist for each MAMS message type. The five MAMS message types are OSS data, stored OSS data, housekeeping data, text message data, and HiRAP data [4]. Regardless of the message type, the format of the individual database tables is shown below.

Table Format

Column 1: Time Extracted from the data packet

Column 2: Payload portion of data packet (MAMS data)

Column 3: Type of data packet (SAMS-II, MAMS, or GSE data)

Column 4: Header portion of the data packet

The data source for the individual MAMS message type database tables is the ISS data stream. Data is placed into this database by the packetGrabber and is ultimately removed from the database by the packetWriter.

4.3.1.2.1 MAMS OSS Database

The MAMS OSS database is used directly in the archival of the raw MAMS OSS acceleration data. The packetWriter software converts individual acceleration data packets in the MAMS OSS data stream from raw counts to engineering unit data. The resultant converted acceleration data is stored in the acceleration data archives for raw MAMS OSS acceleration data. The data source for

the raw MAMS OSS acceleration database is the ISS data stream. Data is placed into this database by the packetGrabber and is ultimately removed from the database by the packetWriter.

4.3.1.2.2 MAMS HiRAP Acceleration Database

The HiRAP database is used as the input for the packetWriter when archiving MAMS HiRAP data. The packetWriter software converts individual acceleration data packets in the MAMS HiRAP data stream from raw counts to engineering unit data. The resultant converted acceleration data is stored in the acceleration data archives for HiRAP data. The data source for the MAMS HiRAP acceleration database is the ISS data stream. Data is placed into this database by the packetGrabber and is ultimately removed from the database by the packetWriter.

4.3.1.3 GSE Packet Data Database

A record in the GSE packet data database has four columns of data extracted from the original UDP data packet received by the packetGrabber. In general, PIMS will utilize a single GSE packet definition for ISS operations. However, the GSE packet generation software in the EHS allows for up to eight different GSE packet definitions. In the event multiple GSE packet definitions are used, a separate database table will be defined for each such GSE packet definition.

Table Format

Column 1: Time Extracted from the data packet

Column 2: Payload portion of data packet (GSE data)

Column 3: Type of data packet (SAMS-II, MAMS, or GSE data)

Column 4: Header portion of the data packet

The data source for the GSE packet database is an output of the ISS data stream processed within the EHS system at MSFC. Data is placed into this database by the packetGrabber and is ultimately removed from the database by the MOIDP.

4.3.2 Interim Databases

There are five interim databases required for generation of PIMS archives. Sections 3.4.3 and 3.4.4 provide an explanation regarding the need for these interim databases. Four of the five interim databases are described in this section. Since bias data is an ancillary data parameter, the MAMS OSS bias database is specifically described in Section 4.3.3.4. All ancillary databases like bias are described in general in Section 4.3.3.

The specialized processing that must be performed to generate the four interim databases described in this section has been described in Section 3.4.4.2. This section is intended to describe the specific definitions for each of the columns in the various interim databases. For each interim database discussed, a regular and/or an artificial database table format can exist. A regular database is one containing specific columns for each parameter in the database (for

example, a time column and 3 acceleration data columns). An artificial packet database is one where two columns exist, one column for time and one column for the data as described in Table 7. Table 6 lists all the archive data types and the database type (real-time or interim) used for generating the archive data files.

4.3.2.1 GSE Packet Data Interim Databases

4.3.2.1.1 Regular GSE Packet Data Interim Database

A record in the regular GSE packet data interim database contains data processed from the original GSE packet data real-timed database. The data are used during real-time mapping of the MAMS OSS data to alternate locations within the ISS. The column definitions for this interim database follow:

Table Format

Column 1: Time Extracted from the GSE data packet

Column 2: Rates and Angles Data Reading #1

Column 3: Rates and Angles Data Reading #2

Column N: Rates and Angles Data Reading #N-1

The data source for the regular GSE packet interim database is the GSE packet real-time database. Data is placed into this database by the MOIDP and also removed from the database by the MOIDP.

4.3.2.1.2 Artificial Packet GSE Packet Data Interim Database

A record in the artificial packet GSE packet data interim database has two columns of data processed from the original GSE packet data real-timed database. The offset to the start of GSE packet data is the default 28 bytes. The data are used in the generation of the archive of GSE packet data. The column definitions for this interim database follow:

Table Format

Column 1: Time Information

Column 2: GSE Data Artificial Packet

The data source for the artificial packet GSE packet interim database is the regular GSE packet interim database. Data is placed into this database by the MOIDP and is ultimately removed from the database by the packetWriter software. The artificial packet subtype definition for GSE packet data is 0x04.

4.3.2.2 Filtered, Non-Bias Compensated MAMS OSS Acceleration Data

This database is an interim database used as the input for the process that bias compensates the MAMS OSS acceleration data enmasse after a given on/off cycle. The MAMS OSS ground processing software converts the MAMS OSS data stream from raw counts to engineering unit

data. The same ground processing software applies the desired filtering (typically TMF) and places the resulting data in this database. The data stored in this database are the filtered, non-bias compensated MAMS OSS data.

Table Format

Column 1: Time Extracted from the data packet

Column 2: X-Axis Acceleration (filtered, but no bias compensation)

Column 3: Y-Axis Acceleration (filtered, but no bias compensation)

Column 4: Z-Axis Acceleration (filtered, but no bias compensation)

The data source for the filtered, non-bias compensated MAMS OSS acceleration data is the MAMS OSS real-time database. Data is placed into this database by the MOIDP and is ultimately removed from the database by the MOIDP at the completion of an on/off cycle.

4.3.2.3 Final Filtered, Bias Compensated MAMS OSS Acceleration Data

This database is an artificial packet interim database used as the input for the packet writer when archiving the final set of MAMS OSS filtered and bias compensated data. The offset to the start of the final filtered, bias compensated MAMS OSS acceleration data is the default 28 bytes. At the completion of an on/off cycle, the MOIDP bias compensates the data from Section 4.3.2.2. The data stored in this database are the final filtered, bias compensated MAMS OSS data.

Table Format

Column 1: Time Information

Column 2: Final Filtered MAMS OSS Data Artificial Packet

The data source for the filtered, bias compensated MAMS OSS acceleration data is the interim database described in Section 4.3.2.2. Data is placed into this database by the MOIDP and is ultimately removed from the database by the packetWriter software. The artificial packet subtype definition for final filtered, bias compensated MAMS OSS acceleration data is 0x02.

4.3.2.4 Best Available Filtered, Bias Compensated MAMS OSS Acceleration Data

This database is an artificial packet interim database used as the input for the packet writer when archiving the “best available” set of MAMS OSS filtered and bias compensated data. The offset to the start of the best available filtered, bias compensated MAMS OSS acceleration data is 40 bytes. This is a 12 byte offset from the default 28 bytes to accommodate the bias x,y,z triplet that is included prior to the acceleration data. The MOIDP software converts the data from Section 4.3.2.2 into bias compensated MAMS OSS data using the bias data available at the time the data is processed. This data is processed after 24 hours, when PIMS has declared that all data has been received. The data stored in this database are the filtered, bias compensated MAMS OSS data.

Table Format

Column 1: Time Information

Column 2: Best Available Filtered MAMS OSS Data Artificial Packet

The data source for the filtered, bias compensated MAMS OSS acceleration data is the interim database described in Section 4.3.2.2. Data is placed into this database by the MOIDP and is ultimately removed from the database by the packetWriter software. The artificial packet subtype definition for best available filtered, bias compensated MAMS OSS acceleration data is 0x01.

4.3.3 Ancillary Data Databases

The databases described in this section are utilized to contain the contents of the ancillary data associated with each acceleration data archive file. Section 3.4.1 lists the various ancillary database parameters required for each ancillary data file. Inputs to the ancillary databases will be made using the Database Interface Tool described in Section 4.8.

4.3.3.1 Data Quality Measure (DQM) Database

The DQM is a text database that describes the "goodness" of the data contained in the file.

Table Format

Column 1:Time

Column 2: Sensor

Column 3:DQM text

PIMS personnel will make inputs to this database when DQM assessments of the acceleration data for particular time intervals have been established. PIMS personnel shall utilize the database interface tool to input DQM data.

4.3.3.2 ISS Configuration Database

This database is a text database that describes the current configuration of the ISS. Modifications to this database will be done as required based on flight number. The increment number associated with a given flight must be included in the configuration text. An example entry would be "Increment: 2, Flight: 6A". The flight information gives an idea of the hardware and equipment on the ISS and the increment number gives information about the crew present. Since the ISS configuration is identical for each sensor, a record in the ISS configuration database has the following form.

Table Format

Column 1:Time

Column 2:ISS Configuration Text

PIMS personnel will make inputs to this database when the ISS configuration changes, either at increment boundaries (crew changes) or at flight boundaries (additional equipment). PIMS personnel shall utilize the database interface tool to input the ISS configuration data.

4.3.3.3 SAMS-II Bias Database

The bias database is a database that contains the numbers used to bias compensate the acceleration data. This database is used to store the various SAMS-II sensors' bias values. MAMS OSS bias data are stored in another database called MAMS OSS bias database. For the SAMS-II, the data is bias compensated on board the ISS. The numbers used by the SAMS-II flight software will be entered for the various SAMS-II flight sensors.

Table Format

Column 1:Time

Column 2: Sensor

Column 3:X-axis Bias

Column 4:Y-axis Bias

Column 5:Z-axis Bias

Inputs to this database will be made by PIMS personnel using information supplied by the SAMS-II program. These inputs will be made when the bias numbers for a particular SAMS-II sensor become available or when the bias numbers have been determined to have changed. PIMS personnel shall utilize the database interface tool to input SAMS-II bias data.

4.3.3.4 MAMS OSS Bias Databases

The MOIDP stores bias numbers generated from the MAMS OSS data stream in these databases. There are four interim bias databases utilized by the MOIDP. Three of these databases are regular interim data bases and one is an artificial packet database.

4.3.3.4.1 MAMS OSS As-Received Bias Database

The first set of numbers in this database is the as-received bias numbers from the MAMS OSS data stream. The as-received bias numbers are the bias numbers calculated from the MAMS OSS data stream when the data stream indicated a bias operation was in progress. The temperature is the MAMS OSS base temperature. This database is specific to only the MAMS OSS data stream.

Table Format

Column 1:Time

Column 2:As-received X-axis Bias

Column 3:As-received Y-axis Bias

Column 4:As-received Z-axis Bias

Column 5:MAMS OSS Base Temperature

The data source for the as-received bias is the MAMS OSS data stream. Data is placed into this database by the MOIDP and also removed from the database by MOIDP.

4.3.3.4.2 MAMS OSS Curve Fit Bias Database

The first set of numbers in this database is the curve fit bias numbers, calculated from the as-received bias data stored in the MAMS OSS As-Received Bias Database. Each time a new bias

point is added to the as-received bias database, the entire curve fit bias database is updated to reflect the latest curve fit information generated using all the bias data received during the current on/off cycle. The temperature is the MAMS OSS base temperature. This database is specific to only the MAMS OSS data stream.

Table Format

Column 1:Time

Column 2:Curve Fit X-axis Bias

Column 3:Curve Fit Y-axis Bias

Column 4:Curve Fit Z-axis Bias

Column 5:MAMS OSS Base Temperature

The data source for the as-received bias database is the MAMS OSS As-Received Bias Database. Data is placed into this database by the MOIDP and also removed from the database by MOIDP.

4.3.3.4.3 Regular MAMS OSS Merged Bias Database

The bias numbers stored in this database represent the merger of the databases described in Sections 4.3.3.4.1 and 4.3.3.4.2. This merger occurs when a MAMS on/off cycle has occurred. This database is specific to only the MAMS OSS data stream.

Table Format

Column 1:Time

Column 2:As-received X-axis Bias

Column 3:As-received Y-axis Bias

Column 4:As-received Z-axis Bias

Column 5:Curve Fit X-axis Bias

Column 6:Curve Fit Y-axis Bias

Column 7:Curve Fit Z-axis Bias

Column 8:MAMS OSS Base Temperature

The data source for the regular merged bias database is the MAMS OSS As-Received Bias Database and the MAMS OSS Curve Fit Bias Database. Data is placed into this database by the MOIDP and also removed from the database by MOIDP.

4.3.3.4.4 Artificial Packet MAMS OSS Merged Bias Database

The bias numbers stored in this artificial packet database are the artificial packetized version of the database described in Section 4.3.3.4.3. The conversion from regular format to artificial packet format is performed to allow packetWriter to generate a complete bias file by processing the bias data enmasse for the on/off cycle. This database is specific to only the MAMS OSS data stream.

Table Format

Column 1: Time Information

Column 2: Merged Bias DataArtificial Packet

The data source for the artificial packet merged bias database is the regular merged bias database. Data is placed into this database by the MOIDP and is removed from the database by packetWriter. The artificial packet subtype definition for MAMS OSS merged bias data is 0x03.

4.3.3.5 Scale Factor Database

The scale factor database is a database that contains the numbers used to scale factor compensate the acceleration data. This database is used to store the various SAMS-II sensors' scale factor values. For the SAMS-II, the data is scale factor compensated on board the ISS. The numbers used by the SAMS-II flight software will be entered for the various SAMS-II flight sensors. For MAMS data, the scale factor numbers will be known pre-flight and will be entered in this database. The scale factor data for both SAMS-II and MAMS are included as part of the ancillary data for completeness. There are no plans to utilize this information in any nominal ground processing.

Table Format

Column 1:Time

Column 2: Sensor

Column 3:Scale Factor X-axis

Column 4:Scale Factor Y-axis

Column 5:Scale Factor Z-axis

The source of the data for the scale factor data will be the SAMS-II and MAMS programs. PIMS personnel shall utilize the database interface tool to input scale factor data.

4.3.3.6 Coordinate System Database

The ancillary data parameters orientation and location are obtained from this database. This database is further utilized by the packet writer software in converting acceleration data from the as-received sensor coordinate system to the Space Station Analysis (SSA) coordinate system. For this purpose, this database works in conjunction with the Data Coordinate System database. The conversion of the data from its as-received sensor coordinate system to a different coordinate system can be done for both the display software and for the archived acceleration data files. The information required to perform this coordinate system conversion is contained in this database.

The offline processing system will utilize this database to perform any coordinate transformations or mapping applications required during the offline processing of the data.

Table Format

Column 1:Time

Column 2: Coordinate System Name (text)

Column 3: Orientation Information (Roll)

Column 4: Orientation Information (Pitch)

Column 5: Orientation Information (Yaw)

Column 6: Location Information (x)

Column 7: Location Information (y)

Column 8: Location Information (z)

Column 9: Location Description

The coordinate system name is a text string that identifies a coordinate system or a current sensor location. Each sensor will have an entry in this table based on its sensor ID. The orientation information for each sensor is responsible for describing the orientation of the sensor coordinate system with respect to the PIMS reference coordinate system (SSA coordinate system as described in [5]. For example, this could be to allow conversion of Sensor 1's data from sensor 1 coordinates to US Lab coordinates. The location information for each sensor is the physical location of the sensor relative to the SSA coordinate system origin in inches. The orientation information for a sensor is an Euler transformation in a yaw, pitch, roll sequence of rotations relative to the SSA coordinate system in degrees. For each sensor, as a minimum, the location description is of the form: <ISS location code>, <free text>. The ISS location code describes the module, rack, and location in the rack, following the location coding system that is described in SSP30575 [6].

Additional entries can be made to this database. For such additional entries, the orientation information is responsible for describing the orientation of coordinate system X with respect to the PIMS reference coordinate system (SSA coordinate system). For example, this could be to allow conversion of data from US Lab coordinates to PCS coordinates.

Additional entries made to this database can serve two purposes. Again, the first purpose is to allow transformation of the data from the SSA coordinate system to coordinate system X. The second purpose would be to input location data for mapping of quasi-steady data to an experiment location. A single entry could serve both purposes.

The source of the data would be PIMS personnel, the SAMS-II program, or the MAMS program. PIMS personnel will utilize the database interface tool to input coordinate system data.

4.3.3.7 Data Coordinate System Database

This database is one of the ancillary databases utilized by the packet writer to generate the header portion of the archived acceleration data file. An entry in this database instructs the packet writer software to write the archived acceleration data file in a coordinate system other than the as-received sensor coordinates. This entry must match an entry made to the Coordinate System

Database described previously. If the coordinate system is a sensor, the most recent entry in the Coordinate System Database will be used.

A record in the data coordinate system database has the form:

Column 1: Time

Column 2: Sensor

Column 3: Data Coordinate System Name (text)

The source of the data would be PIMS personnel, the SAMS-II program, or the MAMS program. PIMS personnel will utilize the database interface tool to input coordinate system data.

4.3.3.7.1 Coordinate System Database and Data Coordinate System Database Examples

Four sample entries for the coordinate system database are provided in Table 12 and three examples of the utilization of the data in this database are provided. Example 1 involves transforming the data from 121f06 sensor coordinates to PCS coordinates. For such a transformation, only the orientation data in columns 3,4, and 5 are relevant. In order for sensor 121f06 to be transformed to the PCS coordinates, the latest entry in the Data Coordinate System database for sensor 121f06 would be:

time	121f06	USLab
------	--------	-------

The real time software and the packetWriter software would combine the orientation matrices to produce a 3x3 matrix that would convert the data directly from sensor coordinates to PCS coordinates. The entry in Table 12 for 121f06 provides the information on transforming the data from the PIMS reference coordinate system (SSA coordinate system) to the 121f06 coordinate system (sensor coordinates). The entry in Table 12 for PCS provides the information on transforming the data from the PIMS reference coordinate system (SSA coordinate system) to the PCS coordinates system.

Example 2 involves only mapping the quasi-steady sensor data to the PCS location. No coordinate system transformation is applied. For this example, the latest entry in the Data Coordinate System database for sensor 'oss' would be:

time	oss	oss
------	-----	-----

Such an entry instructs the real time and packetWriter software to write the MAMS OSS data in MAMS OSS coordinates. Recall, the packetWriter only archives MAMS OSS data at the MAMS OSS location. Therefore, no mapping instructions are needed for the packetWriter. The configuration file used for launching the MAMS OSS application would contain the instruction to map the data to the PCS location (see Section 4.4.2).

Example 3 involves mapping the quasi-steady sensor data to the PCS location and transforming that data into PCS coordinates. For this example, the latest entry in the Data Coordinate System database for sensor ‘oss’ would be:

time	oss	PCS
------	-----	-----

Such an entry instructs the real time and packetWriter software to write the MAMS OSS data in PCS coordinates. The real time software and the packetWriter software would combine the orientation matrices to produce a 3x3 matrix that would convert the data directly from sensor coordinates to PCS coordinates. Again, the configuration file used for launching the MAMS OSS application would contain the instruction to map the data to the PCS location (see Section 4.4.2).

Time	Sensor ID	Orientation	Location	Location Text
T1	121f06	0,0,90	100,200,300	LAB101_K2, PIMS Spare
T2	USLab	90,0,90	100,200,100	LAB101_K2
T3	oss	0,0,180	100,200,50	LAB102_K2
T4	PCS	180,0,90	100,200,300	LAB101_K2, EXPRESS Rack 2

Table 12 - Coordinate System Database Example Entries

4.4 Configuration Files

This section is intended to describe the configuration files necessary to launch the various time domain and frequency domain plots. The contents of these files will be viewable in the monitor and control application (see section 4.5) and can be edited/saved as required to make specialized plots. The contents of these configuration files are composed of user inputs and program inputs. Their relationship is simple: user inputs are employed to calculate program inputs. User inputs are provided in standard text while calculated program inputs are shown in CAPITAL LETTERS. Calculated inputs can be the result of a numeric process or they can be the result of the configuration program automatically supplying information. As necessary during the description of the contents of each configuration file type, individual parameters will be explained in detail.

Each display type will have a default configuration file and N specialized configuration files. The default files are intended to provide a consistent look to publicly available plots for different heads being displayed using the same display type (ie two spectrograms should have identical color limits, etc. when displayed on the public PIMS web page). The specialized configuration files will be used by analysts as required and will be displayed on a private PIMS web page.

For each display, two configuration files are described: user input configuration file and run configuration file. For a given display, there will exist N user input configuration files, one of which will be termed the default. At the time a display is launched, the user input configuration files that was selected will be utilized to generate the run configuration file. This run

configuration file will get re-generated each time a display is initiated. The reason for this relationship is that the run configuration file must use the user input configuration file and any required databases to "calculate" all required parameters. Those "calculated" parameters must be checked each time a display is initiated.

Both the user input configuration files and the run configuration files will contain information about the expected units for the individual configuration file elements. For example,
 snap_interval (seconds): 5
 Both configuration file types need to clearly indicate the expected units of an individual file element. The organization of both configuration file types will be identical, consisting of all user input parameters followed by any calculated parameters. The first calculated parameter in the file will always be COORD_SYSTEM.

The format of the files consists of generic parameters and plot specific parameters. A basic entry in the configuration file is:

Parameter (Parameter Units): Parameter Value

The generic parameters for the various configuration files are described in Table 13.

Parameter	Parameter Units	Parameter Description
screen_snap	0=OFF, 1=ON	Indicates whether JPG snapshots of displays should be active
snap_interval	Seconds	Time interval in seconds between JPG snapshots
COORD_SYSTEM	text	Specifies the coordinate system for displays
CONVERT_COORD	Matrix	1 x 9 matrix required to convert data from sensor coordinate system to the coordinate system specified by COORD_SYSTEM
head_start	minutes	Amount of time to display at activation. Head_start=5, instructs the display software to start display using data that are 5 minutes old
database	text	Database name (default=pims)
jpg_path	text	Where to send the JPG snapshots (default=pims@tsscruader.grc.nasa.gov:/sdds/pims/www/plots)
jpg_quality	percentage	Define the quality of the JPG snapshots
axis_select	Text	Select axis to plotting
TABLE_LOCATION	Text	Location of UDP Packet database
TABLE_NAME	Text	Name of the UDP Packet database

Table 13 - Configuration File Generic Parameters

COORD_SYSTEM is selected by the user at the same time sensor, application, and configuration file are selected. CONVERT_COORD is calculated from data present in the Coordinate System Database based on the value of COORD_SYSTEM. Again, all calculated parameters will be displayed in CAPITALS. The text string specified in COORD_SYSTEM directs the configuration file program to the proper coordinate system conversion values to go from a sensor coordinates (the as-received form) to the coordinate system specified by the parameter COORD_SYSTEM. Section 4.3.3.6 provides details regarding the Coordinate System Database.

The contents of the configuration files will be separated by the type of display supported. The three types of displays are: Time Domain Configuration Files (SAMS-II or HiRAP), MAMS OSS Time Domain Configuration Files, and Frequency Domain Configuration Files (SAMS-II or HiRAP). Each display type will have the name of the corresponding executable and two lists of parameters. The first list is called the user input configuration file. The parameters in this list consist of the desired parameters for the selected display. The second list will be referred to as the run configuration file. The parameters in this list consist of the actual parameters for the selected display. For example, for a color spectrogram, a user may input in the user input configuration file an entry of $\text{delta}_f=0.05$ Hz. This is the desired frequency resolution. Based on the other parameter values, the run configuration file may provide a "calculated" $\text{delta}_f=0.044$ Hz. This represents the actual frequency resolution used for the spectrogram display. Again, "calculated" means either a numeric calculation or data derived from the user inputs selected. For example, the parameter MAPPING_LOCATION is obtained from Coordinate System Database (Section 4.3.3.6) based on the value of the user input `mapping_site`.

4.4.1 Time Domain Configuration Files (SAMS-II or HiRAP)

4.4.1.1 Acceleration vs. Time

Executable Name: gvt.sams, gvt.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	mg's-min,max,tick	gscale [#]	mg's-min,max,tick
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
SAMPLINGRATE	Hz	SAMPLINGRATE	Hz

- gscale consists of 3 parameters: gmin, gmax, and tick. Setting tick=0, enables Labview default

4.4.1.2 Interval Minimum/Maximum Acceleration vs. Time

Executable Name: minmax.sams, minmax.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	mg's-min,max,tick	gscale [#]	mg's-min,max,tick
interval	Seconds	interval	Seconds
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
SAMPLINGRATE	Hz	SAMPLINGRATE	Hz
NI	Samples	NI	Samples

- gscale consists of 3 parameters: gmin, gmax, and tick: Setting tick=0, enables Labview default

4.4.1.3 Interval Average Acceleration vs. Time

Executable Name: intavg.sams, intavg.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	mg's-min,max,tick	gscale [#]	mg's-min,max,tick
interval	Seconds	interval	Seconds
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
SAMPLINGRATE	Hz	SAMPLINGRATE	Hz
NI	Samples	NI	Samples

- gscale consists of 3 parameters: gmin, gmax, and tick: Setting tick=0, enables Labview default

4.4.1.4 Interval RMS Acceleration vs. Time

Executable Name: intrms.sams, intrms.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	mg's-min,max,tick	gscale [#]	mg's-min,max,tick
interval	Seconds	interval	Seconds
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
SAMPLINGRATE	Hz	SAMPLINGRATE	Hz
NI	Samples	NI	Samples

- gscale consists of 3 parameters: gmin, gmax, and tick: Setting tick=0, enables Labview default

4.4.2 Time Domain Configuration Files (MAMS OSS)

4.4.2.1 Acceleration vs. Time

Executable Name: ossgvt.oss

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	μg's-min,max,tick	gscale [#]	μg's-min,max,tick
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
HOST_FOR_BIAS	Text	HOST_FOR_BIAS	Text

- gscale consists of 3 parameters: gmin, gmax, and tick. Setting tick=0, enables Labview default

4.4.2.2 Interval Minimum/Maximum Acceleration vs. Time

Executable Name: ossminmax.oss

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	μg's-min,max,tick	gscale [#]	μg's-min,max,tick
interval	Seconds	interval	Seconds
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
HOST_FOR_BIAS	Text	HOST_FOR_BIAS	Text
NI	Samples	NI	Samples

1. # - gscale consists of 3 parameters: gmin, gmax, and tick: Setting tick=0, enables Labview default
2. @ - Mapping_site is a text string that must correspond to an entry in the Coordinate System Database

The parameter MAPPING_LOCATION is the x,y,z coordinates of the location specified by the parameter Mapping_site. Multiple mapping site and mapping locations can be specified in a given configuration file.

4.4.2.3 Interval Average Acceleration vs. Time

Executable Name: ossintavg.oss

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	μg's-min,max,tick	gscale [#]	μg's-min,max,tick
interval_size	Seconds	interval_size	Seconds
interval_step	Seconds	interval_step	Seconds
mapping_site [@]	Text	mapping_site	Text
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
HOST_FOR_BIAS	Text	HOST_FOR_BIAS	Text
MAPPING_LOCATION	Inches – x, y, z	MAPPING_LOCATION	Inches – x, y, z
NI	Samples	NI	Samples

- # - gscale consists of 3 parameters: gmin, gmax, and tick: Setting tick=0, enables Labview default
- @ - Mapping_site is a text string that must correspond to an entry in the Coordinate System Database

The parameter MAPPING_LOCATION is the x,y,z coordinates of the location specified by the parameter Mapping_site. Multiple mapping site and mapping locations can be specified in a given configuration file.

TMF Acceleration vs. Time

Executable Name: tmf.oss

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
timespan	Minutes	timespan	Minutes
gscale [#]	μg's-min,max,tick	gscale [#]	μg's-min,max,tick
TMF_FilterSize	Seconds	TMF_FilterSize	Seconds
TMF_FilterStep	Seconds	TMF_FilterStep	Seconds
mapping_site [@]	Text	mapping_site	Text
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
HOST_FOR_BIAS	Text	HOST_FOR_BIAS	Text
MAPPING_LOCATION	Inches – x, y, z	MAPPING_LOCATION	Inches – x, y, z

1. # - gscale consists of 3 parameters: gmin, gmax, and tick. Setting tick=0, enables Labview default
2. @ - Mapping_site is a text string that must correspond to an entry in the Coordinate System Database

The parameter MAPPING_LOCATION is the x,y,z coordinates of the location specified by the parameter Mapping_site. Multiple mapping site and mapping locations can be specified in a given configuration file. The parameter TMF_FilterSize is the number of seconds to consider for the TMF process and the parameter TMF_FilterStep is the number of seconds between TMF data points. For example, the parameter settings of TMF_FilterSize=48, TMF_FilterStep=16 consider 48 seconds of data for the generation of each TMF data point. The window slides by 16 seconds to calculate the next TMF data point.

4.4.3 Frequency Domain Configuration Files (SAMS-II and HiRAP)

4.4.3.1 PSD vs. Frequency

Executable Name: psd.sams, psd.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
axis_select	Text (x,y,z,combined)	axis_select	Text (x,y,z,combined)
timespan	Minutes	timespan	Minutes
delta_f	Hertz	delta_f	Hertz
fscale [#]	Hz-min,max,tick	fscale	Hz-min,max,tick
windowing	Text	windowing	Text
spectralvariance	Percent	spectralvariance	Percent
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
TIMEINTERVAL	Seconds	TIMEINTERVAL	Seconds
SAMPLINGRATE	Hertz	SAMPLINGRATE	Hertz
NFFT	Samples	NFFT	Samples
NO	Samples	NO	Samples
P	Percent	P	Percent
K	# of spectral averages	K	# of spectral averages

- fscale consists of 3 parameters: min, max, and tick Setting tick=0, enables Labview default

$\log_2(\text{NFFT})$ is a positive integer and spectral variance is defined such that $0 < \text{spectralvariance} \leq 100\%$

The parameter SAMPLINGRATE is extracted from the data stream. Setting the parameters delta_f and temporalresolution equal to zero is the “fullnes” flag for spectrograms. Setting either delta_f or temporalresolution equal to zero requests that the parameter be calculated based on the other user inputs (delta_f=0 requests delta_f be calculated based on the value of temporal resolution; effectively a “don’t care”).

4.4.3.2 PSD vs. Time vs. Frequency (Color Spectrogram)

Executable Name: spec.sams, spec.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
axis_select	Text (x,y,z,combined)	axis_select	Text (x,y,z,combined)
timespan	Minutes	timespan	Minutes
delta_f	Hertz	delta_f	Hertz
fscale [#]	Hz-min,max,tick	fscale	Hz-min,max,tick
temporalresolution	Seconds	temporalresolution	Seconds
windowing	Text	windowing	Text
spectralvariance	Percent	spectralvariance	Percent
colorbarlimits	g ² /Hz – min,max	colorbarlimits	g ² /Hz – min,max
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
TIMEINTERVAL	Seconds	TIMEINTERVAL	Seconds
SAMPLINGRATE	Hertz	SAMPLINGRATE	Hertz
NFFT	Samples	NFFT	Samples
NO	Samples	NO	Samples
P	Percent	P	Percent
K	# of spectral averages	K	# of spectral averages

- fscale consists of 3 parameters: min, max, and tick. Setting tick=0, enables Labview default

log₂(NFFT) is a positive integer and spectral variance is defined such that 0 < spectralvariance ≤ 100%

The parameter SAMPLINGRATE is extracted from the data stream. Setting the parameters delta_f and temporalresolution equal to zero is the “fullness” flag for spectrograms. Setting either delta_f or temporalresolution equal to zero requests that the parameter be calculated based on the other user inputs (delta_f=0 requests delta_f be calculated based on the value of temporal resolution; effectively a “don’t care”).

4.4.3.3 Cumulative RMS vs. Frequency

Executable Name: cumrms.sams, cumrms.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
axis_select	Text (x,y,z,combined)	axis_select	Text (x,y,z,combined)
timespan	Minutes	timespan	Minutes
delta_f	Hertz	delta_f	Hertz
fscale [#]	Hz-min,max,tick	fscale	Hz-min,max,tick
windowing	Text	windowing	Text
spectralvariance	Percent	spectralvariance	Percent
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
TIMEINTERVAL	Seconds	TIMEINTERVAL	Seconds
SAMPLINGRATE	Hertz	SAMPLINGRATE	Hertz
NFFT	Samples	NFFT	Samples
NO	Samples	NO	Samples
P	Percent	P	Percent
K	# of spectral averages	K	# of spectral averages
k	unitless freq bin index	k	unitless freq bin index

- fscale consists of 3 parameters: min, max, and tick. Setting tick=0, enables Labview default

log₂(NFFT) is a positive integer and spectral variance is defined such that 0 < spectralvariance ≤ 100%

The parameter SAMPLINGRATE is extracted from the data stream. Setting the parameters delta_f and temporalresolution equal to zero is the “fullness” flag for spectrograms. Setting either delta_f or temporalresolution equal to zero requests that the parameter be calculated based on the other user inputs (delta_f=0 requests delta_f be calculated based on the value of temporal resolution; effectively a “don’t care”).

4.4.3.4 RMS Acceleration for Selected Frequency Bands

Executable Name: rms.sams, rms.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
axis_select	Text (x,y,z,combined)	axis_select	Text (x,y,z,combined)
rmsscale	gRMS-min,max,tick	rmsscale	gRMS-min,max,tick
frequencybands	Hz-f1,f2,...f10	frequencybands	Hz-f1,f2,...f10
delta_f	Hertz	delta_f	Hertz
windowing	Text	windowing	Text
timespan	Minutes	timespan	Minutes
temporalresolution	Seconds	temporalresolution	Seconds
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
TIMEINTERVAL	Seconds	TIMEINTERVAL	Seconds
SAMPLINGRATE	Hertz	SAMPLINGRATE	Hertz
NI	Samples	NI	Samples
NFFT	Samples	NFFT	Samples
k	unitless freq bin index	k	unitless freq bin index
NO	Samples	NO	Samples
P	Percent	P	Percent

$\log_2(\text{NFFT})$ is a positive integer and spectral variance is defined such that $0 < \text{spectralvariance} \leq 100\%$

The parameter SAMPLINGRATE is extracted from the data stream. Setting the parameters delta_f and temporalresolution equal to zero is the “fullness” flag for spectrograms. Setting either delta_f or temporalresolution equal to zero requests that the parameter be calculated based on the other user inputs (delta_f=0 requests delta_f be calculated based on the value of temporal resolution; effectively a “don’t care”). The parameter frequencybands define selected frequency bands of interest. F1 is the lower frequency. An input of 10,12,16,18 would calculate RMS acceleration for the frequency bands 10-12 Hz and 16-18 Hz. A maximum of five frequency band pairs can be input. The maximum frequency in the frequency bands list cannot exceed the cutoff frequency.

4.4.3.5 RMS Acceleration vs. One-Third Octave Frequency Bands

Executable Name: octave.sams, octave.hirap

User Input Configuration File		Run Configuration File	
Parameter (User Input)	Parameter Units	Parameter (Calculated)	Parameter Units
screen_snap	0=OFF/1=ON	screen_snap	0=OFF/1=ON
snap_interval	Seconds	snap_interval	Seconds
head_start	Minutes	head_start	Minutes
database	Text	database	Text
jpg_path	Text	jpg_path	Text
jpg_quality	Percentage	jpg_quality	Percentage
axis_select	Text (x,y,z,combined)	axis_select	Text (x,y,z,combined)
Mode	Text (0,1, or 2)	mode	Text (0,1, or 2)
timespan	Minutes	timespan	Minutes
Curveoverlay	(0=OFF/1=ON)	curveoverlay	(0=OFF/1=ON)
Temporalresolution	Seconds	temporalresolution	Seconds
Rmsscale	gRMS-min,max,tick	rmsscale	gRMS-min,max,tick
fscale [#]	Hz-min,max,tick	fscale	Hz-min,max,tick
delta_f	Hertz	delta_f	Hertz
Windowing	Text	windowing	Text
Spectralvariance	Percent	spectralvariance	Percent
COORD_SYSTEM	Text	COORD_SYSTEM	Text
CONVERT_COORD	1x9 matrix	CONVERT_COORD	1x9 matrix
TABLE_LOCATION	Text	TABLE_LOCATION	Text
TABLE_NAME	Text	TABLE_NAME	Text
TIMEINTERVAL	Seconds	TIMEINTERVAL	Seconds
SAMPLINGRATE	Hertz	SAMPLINGRATE	Hertz
NFFT	Samples	NFFT	Samples
NO	Samples	NO	Samples
P	Percent	P	Percent
K	# of spectral averages	K	# of spectral averages

- fscale consists of 3 parameters: min, max, and tick. Setting tick=0, enables Labview default

log₂(NFFT) is a positive integer and spectral variance is defined such that 0 < spectralvariance ≤ 100%

The parameter SAMPLINGRATE is extracted from the data stream. Setting the parameters delta_f and temporalresolution equal to zero is the “fullness” flag for spectrograms. Setting either delta_f or temporalresolution equal to zero requests that the parameter be calculated based on the other user inputs (delta_f=0 requests delta_f be calculated based on the value of temporal resolution; effectively a “don’t care”).

The parameter mode is described as follows:

- 0=manual mode, $NFFT = \text{SAMPLINGRATE} / \text{delta_f}$
- 1="100 second mode", $NFFT = \text{ceil}(100 * \text{SAMPLINGRATE})$
- 2="power of 2", $NFFT = 2^{\text{ceil}(\log_2(100 * \text{SAMPLINGRATE}))}$

4.5 Monitor and Control Application

The Monitor and Control application is utilized to launch and terminate display applications on the various machines dedicated to processing the acceleration data. Configuration files can be edited and saved using this application. The application is written using Labview and is executed from the monitor and control computer in the TSC.

A display application for a given sensor can be started or terminated by selecting the sensor of interest from the main table. This activates the appropriate additional tables required to properly start a display application for that sensor type. The mapping location (MAMS OSS data only), the coordinate system, and the application are selected after the sensor is selected from the main table. Selection of the application populates the configuration files available for initiating that particular application. For example, a color spectrogram configuration file will not be provided as a configuration file option when an interval minimum/maximum display has been selected. The final window allows the user to launch the application on the local desktop of the host machine or to launch the application within a Virtual Network Computer (VNC) window. For the local desktop option, the destination machine must be logged in as user=pims with screen saver options disabled.

The main table consists of nine columns that describe the sensor data available for processing, whether or not the sensor is actively receiving data, and the sampling rate and corresponding cutoff frequency of that sensor. The main table accesses the ancillary databases for the latest coordinate system and sensor location data for the sensor. An example of the main table is shown in Table 14 and the corresponding additional tables are shown in Table 15.

Host Sensor	Sensor Name	Sensor Status	Sensor Coord.	Sensor Location	Sample Rate	Cutoff Freq.	Packets Rec'd	PW Status
Justice	oss	AOS	oss	LAB1O1, ER #1	10	1	8123	ON
Thome	121f02	Off	121f02	LAB1O1, ER #1	1000	400	0	OFF
Colon	121f03	Off	121f03	LAB1O1, ER #1	500	200	0	OFF
Sexson	121f04	Off	121f04	LAB1O1, ER #1	250	100	0	OFF
Fryman	121f06	AOS	121f06	LAB1O2, ER #2	125	50	11564	ON

Table 14 - Monitor and Control Application Main Table

Coordinate System	Select Application	Select Mapping Location	Config File	Launch App On	VNC Port	Active Apps
oss	gvt	oss	default.tmf	Justice	0	
121f02	minmax	121f02	special.tmf	Thome	1	
121f03	intavg	121f03		Colon	2	
121f04	tmf	121f04		Sexson		
121f06		121f06		Fryman		
PCS		PCS				

Table 15 – Monitor and Control Tables

The highlighted text in Table 14 and Table 15 represent a sample set of user selections. In this case, the user requested a MAMS OSS trimmean filtered acceleration versus time plot be initiated on the processing machine justice. The selection from the main table indicates that the MAMS OSS data stream is active and that the latest input for MAMS OSS data in the data coordinate system database indicates MAMS OSS data will be archived in MAMS OSS coordinates. Since MAMS OSS data was selected from the main table, all the additional monitor and control tables are activated for user input. There are no active MAMS OSS applications. The MAMS OSS coordinate system has been selected as the coordinate system for display of the MAMS OSS data. The desired display type has been selected as the TMF display. In this case, the user selects to map MAMS OSS data to the OSS location and the PCS experiment location. Further, the configuration file selected is the default.tmf configuration file. The display is to be shown on justice’s local desktop.

4.6 packetWriter

The packetWriter application is responsible for generating the t, x, y, z acceleration data archives and their associated ancillary data files. As data stored in the various PIMS databases are processed by packetWriter, the database records are removed from the database. In this way, packetWriter is responsible for maintaining many of the PIMS databases. The MOIDP maintains MAMS specific interim databases. It becomes active in 30 minute cycles and processes data 24 hours after the data has been received. The 24 hour delay is designed to allow all AOS and LOS data to be properly received and merged prior to processing by packetWriter.

4.6.1 General Processing

PacketWriter retrieves data packets from a database, determines the packet type, decodes the data in the packets, and writes the data in standard format data files for archiving and offline processing. The primary purpose of packetWriter is to centralize the decoding of the many packet types present in the PIMS system. Before it data arrives at the PIMS equipment, all of the data

has been broken in to small UDP network packets (as opposed to a continuous stream of data inherent in a TCP network). PacketWriter reconstructs stream-like contiguous data files from packetized data.

There are many reasons why the data may not be contiguous. A common reason is that the ISS is in a LOS state, so no data is arriving. When the station returns to AOS, it will send current data as well as older data queued up during the LOS period. To allow for these LOS data gaps to be filled in, packetWriter will usually delay processing data until it is at least 24 hours old. However, if packetWriter determines that there is contiguous data that extends beyond this threshold, it will continue to process the data until it comes to a gap. Processing of the data after the gap will resume when the data is 24 hours old.

Non-contiguous data may also be caused if a packet is dropped by one of the networks before it arrives in the database. Since the UDP protocol provides for no error detection, correction or retransmission, these packets cannot be recovered unless retransmission of data is requested. The data in a file generated by packetWriter are always contiguous, so when the network drops a packet transmitting the data to PIMS, a new file is started.

A new file may be started for other reasons as well. There is a file size limit (default is 10 minutes) after which a new file will be started. If the file size limit is reached, the next file name will contain a '+' character separating the time span referenced in the file name rather than a '-' character to indicate that this file can be considered contiguous with the previous file. A new file will also be started if any of the information in the header needs to be changed (for example, due to a sample rate change). Appendix C documents the relationship between generating new data files, ancillary data parameters, and the various data types available for processing.

Some of the information in the header files, such as start time and sample rate, come from the data packets themselves. Other information in the header files, such as data quality measure and ISS configuration descriptions, come from a set of ancillary databases. The ancillary information is usually using the database interface tool described in section 4.8. All ancillary databases have a time column that indicates when the database record takes effect. PacketWriter will use the most recent ancillary database record with a timestamp earlier than the packet it is processing. Where applicable, the sensor column in the record must also match the current packet being processed. If there is no ancillary record with an earlier time and matching sensor, then the ancillary data from that particular database will not appear in the header file.

4.6.2 Coordinate Systems

PacketWriter can rotate acceleration data to a new coordinate system before they are written to files. Coordinate system information is controlled by entries in the ancillary databases. The 'data_coord_system' database specifies the name of the coordinate system in which to present the

data for a particular sensor. If no entry is found, packetWriter will default to 'sensor' coordinates (no rotation of data). If an entry is found, packetWriter will look for entries for the sensor coordinates and the presentation coordinates in the 'coord_system_db' database. If either of these are missing (or if both are identical) packetWriter will default to 'sensor' coordinates. If both are found, packetWriter will rotate all xyz acceleration data to the reference coordinate system, then rotate the data from the reference coordinate system to the display coordinate system. Entries describing the 'SensorCoordinateSystem' and 'DataCoordinateSystem' will appear in the header files.

4.6.3 PacketWriter Inputs

PacketWriter's primary input comes from MySQL database tables (created by the packetGrabber software) that contain columns for time and packet. The 'time' column is represented as an 8 byte double floating point number representing the number of seconds since the start of the Unix epoch (since January 1, 1970). The database is keyed on the time column so there is only one possible entry for a given time. The 'packet' column is represented as a 'BLOB', which is essentially a variable length collection of bytes as received from the UDP network packet.

PacketWriter's other inputs come from the ancillary databases, which vary in format but all have columns for time and usually sensor name. PacketWriter finds the most recent entry that matches the sensor name to determine which ancillary data to use.

4.6.4 PacketWriter Outputs

The pad files generated by packetWriter always come in pairs: a binary data file and a text (XML) header file.

The binary data file usually consists of 4 values per sample representing time, acceleration along the x-axis, acceleration along the y-axis, and acceleration along the z-axis. All values are by default 4 byte little endian floating point numbers. There are command line options for changing this to big endian or ASCII representations. Time is measured in seconds and acceleration data are measured in g's. The time of the first sample in a file is always 0.0, and the times of other samples in the file are relative to the first sample. This is because a 4 byte float does not have enough significant digits to hold an absolute time with enough precision to identify individual samples. The absolute time of the first sample is contained in the header file.

The header file is ASCII text written in an XML format. Most entries in the file should be self explanatory, but there are some things to watch out for. Some of the entries come from ancillary databases which are updated by hand (such as "DataQualityMeasure"), and therefore they might contain characters that mean something special to XML. These special characters will be replaced by their XML equivalents according to the following table:

Text Character	XML Equivalent
&	&
<	<
>	>
“	"
‘	'

Table 16 - XML Special Characters

4.6.5 Installation and Operational Requirements

PacketWriter is written in the Python programming language, and requires a Python version 1.5 interpreter to run. This is currently installed by default on Red Hat Linux systems. In addition, packetWriter requires two add-on Python packages that are not installed by default: Numeric Python (the package is actually called 'python-numpy') and a Python interface to the MySQL database (called 'MySQL-python').

Currently, the packetWriter program consists of two files: 'packetWriter.py' and 'accelPacket.py'. These files should be kept together in the same directory on the users path. Because packetWriter creates temporary files and directory structures in the directory in which it is started, packetWriter should not be started twice within the same directory. Also, because packetWriter normally deletes data from the database after processing, you must take care not to process the same database table simultaneously with multiple copies of packetWriter. This simultaneous processing would be allowable if the deletion command line parameter was disabled at run time and if the resulting output were stored in different directories.

A hard-coded user ID ('pims') and password are used to access the MySQL database, and MySQL permissions (specifically, the 'user' table in the 'mysql' database) must be configured to allow access to the user and computer used by packetWriter. Database permissions must be set properly on both the computer hosting the data to be processed, and on the ancillary host computer that holds the ancillary databases (used for coordinate system information, etc).

To move data files to a remote destination, the secure copy (scp) program is used. Secure copy is part of the secure shell package (called "openssh"), which is not installed by default. Openssh requires the installation of Secure Socket Layer support (called openssl). Because packetWriter must be able to transfer files without being prompted for a password, RSA authentication is required to be configured properly for the users and computers involved. If packetWriter is started and a remote destination is specified, packetWriter will test the scp configuration by attempting to copy a dummy file. Execution will abort with an error message if this test fails.

4.6.6 PacketWriter Command Line Parameters

The packetWriter software has several command line parameters that can be utilized at the time packetWriter is invoked. These command line parameters are described with their default values in Table 17. All command line parameters must be set with the 'parameter=value' format, and parameters must be separated by white space. For example, packetWriter could be invoked with a line like this:

```
packetWriter.py ancillaryHost=justice maxFileTime=300
```

Parameter	Default Value	Description
additionalHeader	null string ""	Insert additional text in the header files. Header information is in XML format, and contains characters that are not legal on the command line. To represent these characters, they must be replaced by characters that are legal: a space character must be replaced by '#' a tab character must be replaced by '~' a carriage return character must be replace by '~~'
ancillaryHost	None	Host machine name for the ancillary database information. Default is 'None', which means no ancillary information will appear in the header files.
ascii	0	0=Do not output txyz data in ASCII format 1=Output txyz data in ASCII format (instead of binary)
bigEndian	0	0=Output binary data in little Endian format 1=Output binary data in big Endian format
cutoffDelay	24*3600	Process data after waiting this many seconds. This delay time is subtracted from the current time to determine if data is old enough to process.
database	pims	Database name for the tables to be processed
delete	0	0=do not delete data from database when processed 1=delete data from database as it is processed 'tableName'=move data into 'tableName' before deleting
destination	.	Specify the destination machine and directory for the data files, in secure copy format. 'host:/path/to/data' The default is '.', which means leave the data in the directory where packetWriter was started.
endTime	0.0	Process data with time stamps less than endTime. The

		default of '0.0' means no limit.
startTime	0.0	Process data with time stamps greater than startTime. The default of '0.0' means no limit (any data back to Jan 1, 1970 will be processed).
host	localhost	Host machine name for the database to be processed
maxFileTime	600	Max file size in seconds
quitWhenDone	1	0=packetWriter stays active waiting for additional data if all data in table is processed 1=packetWriter terminates when table is empty
resume	1	1=attempt to pick up processing where it left off (if a previous run of packetWriter was interrupted) 0=process the database from the beginning
showWarnings	1	0=turn processing warnings off 1=show packetWriter processing warnings
tables	ALL	Database table name(s) to process. Table names must be separated with a comma, but no space. The default of 'ALL' means to process all tables that are found with the required 'time' and 'packet' columns.

Table 17 - packetWriter Input Parameters

4.7 PIMS Web Page

The PIMS ISS web page is the primary link between the PIMS project and the microgravity community. The web page consists of links separated into five functional areas: 1) reference links, 2) real time links, 3) data requests, 4) disturbance characterization, and 5) neural network results.

4.7.1 Reference Links

The reference links provide access to the *PIMS home page* and the *Acceleration Measurement Home Page*. Both of these links are top-level references to information about the PIMS project and the various acceleration measurement systems used by the PIMS project for acceleration data analysis.

4.7.2 Real Time Links

The real time links provide access to the ISS acceleration data plots generated by the PIMS real time software systems. Access to the real time plots can be obtained through the *Current Real-Time Plots* link and the *Current Instrument Locations* link. The *Current Real-Time Plots* provides a sequence of menus that allows the user to select the desired plot for display on their local terminal. The first menu allows the selection of the desired acceleration measurement

system. The second menu provides a list of the sensors available for that measurement system. The final menu allows selection of the desired data plot for display in the user's browser.

The *Current Instrument Locations* link allows the same capability as the *Current Real-Time Plots* link, but utilizes the current ISS rack topology as a means of simplifying access to a given accelerometer's data. Selecting the link provides a drawing of the rack topology, with all racks containing PIMS supported accelerometers outlined in green. All payload racks are colored orange. Rolling over a rack location results in a similar set of three menus (measurement system, sensor, and plot types), but only the accelerometers located in the selected rack are displayed as options. Again, selection of the desired data plot results in display of that plot type in the user's browser.

4.7.3 Data Requests

There are three links that fall under the data requests category. These links allow the users to access the PIMS ISS acceleration data archives, to request data plots, and to status their data requests. The link *Access ISS Acceleration Data Archives* allows the user to access the stored t,x,y,z acceleration data files directly for file transfer to the user's local computer.

The *Request Data Plots* and the *Status Data Plots* links work in conjunction to allow the user to request a plot of acceleration data and to status that requested plot.

4.7.4 Disturbance Characterization

The *View Interesting Data Plots* link allows the user access to a compilation of disturbances characterized by the PIMS project. This compilation of interesting data plots will allow the user to view data plots grouped into logical categories such as exercise, turn on/turn off events, and vehicle docking and undocking. Such a logical grouping of known microgravity environment disturbances will aid the user in easily accessing ISS disturbance data.

4.7.5 Neural Network Information

A further simplification of data access is provided by the *ISS Microgravity Monitoring System*. This link provides access to the results of the Microgravity Environment Monitoring System (MEMS). The MEMS processes the acceleration data and outputs a more simplified display summarizing the current microgravity environment. Such a system interprets disturbance features in the data and provides a summary display indicating when various features are active or inactive. The user can examine the simplified display and obtain a straightforward understanding of the current environment.

4.8 Database Interface Tool

The database interface tool provides the means to enter new information into six of the seven ancillary databases described in section 4.3.3. This tool does not provide access to the MAMS OSS bias databases; these are controlled by the MOIDP. The ancillary databases controlled by the

database interface tool are located on the host machine for the Monitor and Control computer. This application is written using Labview.

Inputs to the ancillary databases are accomplished through a series of five pull down menus/input windows. These windows are:

- Select a Table
- Select Sensor
- Enter/Select Coord Name
- Enter/Select ISS Config/Location Name
- Enter/Select DQM Status

There are two additional windows for providing orientation data and location, scale factor, or bias data. When a table is selected, the appropriate set of pull down menus/inputs windows are activated. For example, selecting the ISS configuration database table activates the 'Enter/Select ISS Config/Location Name' window and de-activates the remaining windows. When a database table is selected and the appropriate inputs are provided through these windows and the pull down menus, the database can be updated by using the Selection Operation window and the Insert New Record option. The latest input for a given database will appear at the top of the display.

4.9 MAMS OSS Interim Data Processor

The MOIDP performs all MAMS OSS data processing with the exception of the generation of the raw MAMS OSS acceleration data archives, which is handled directly by packetWriter. The MOIDP is responsible for the generation and maintenance of all the MAMS OSS interim database products (see sections 4.3.2.2, 4.3.2.1.1, 4.3.3.4.1, 4.3.3.4.2, and 4.3.3.4.3) and for the generation and maintenance of all MAMS OSS artificial packet database products (see sections 4.3.3.4.4, 4.3.2.1.2, 4.3.2.3, and 4.3.2.4). Additionally, the MOIDP processes the GSE packet data for use by the MAMS OSS real time applications and for use by the packetWriter in generating GSE packet data archives.

4.9.1 MOIDP Nominal Operations

Nominal operations for the MOIDP consists of retrieving data from the MAMS OSS real time database and updating and maintaining the filtered, non-bias compensated database (section 4.3.2.2), the as-received and curve fit bias databases (sections 4.3.3.4.1 and 4.3.3.4.2), and the best available, filtered, bias compensated database (section 4.3.2.4). When a new MAMS OSS data packet is retrieved from the MAMS OSS real time database, the MOIDP determines whether the data are from a bias interval. If bias data are detected, the bias databases are updated accordingly. If the data packet received contains nominal MAMS OSS data, the data are written to both the the filtered, non-bias compensated database and the best available, filtered, bias compensated data base.

The packetWriter software will output the best available archive data set throughout these operations. The bias data stored in the header portion of the best available archives is a single x,y,z bias triplet. All the data within a given best available data archive were bias compensated using this single bias triplet. The filtered, non-bias compensated data set is accumulated in the database for a given MAMS OSS on/off cycle. No archives of these data are generated until the on/off cycle is detected.

4.9.2 MOIDP MAMS OSS On/Off Operations

A sequence of events occurs when an on/off cycle is detected by the MOIDP. The first operation consists of generating a merged bias database consisting of the as-received bias data and the curve fit of the bias data. This database is described in section 4.3.3.4.3. These bias data are written to the bias data archives by packetWriter. The entire bias data set for a given on/off cycle are contained in a single bias data archive file.

With the entire bias data for a given on/off cycle in a single archive file, the data contained in the filtered, non-bias compensated data (section 4.3.2.2) can be processed en masse using the bias data for the entire on/off cycle. The processing results in the artificial packet final, filtered bias compensated database described in section 4.3.2.3. PacketWriter is invoked by the MOIDP to generate the archive data files. The resultant archives are the MAMS OSS filtered, bias compensated data set. This data set represents the most accurate filtered MAMS OSS archive data since it was generated using all the bias data for a given on/off cycle.

4.10 Data Recovery and Gap Detection Applications

4.10.1 SDDS Gap Detector

This application is written in Labview and is designed to detect gaps between PAD files located on the SDDS that are greater than a user provided duration. The purpose of this application is to examine the contents of a particular directory on the SDDS and allow the user to determine if significant data gaps exist. After packetWriter processes data and generates an archive data file for a given data type, this application can be executed to determine whether there are any time gaps between data files. The user provides the packetWriter command line parameters as discussed in section 4.6.6. Additionally, the user provides the directory to be examined and the gap criteria. The gap criteria is a time entry in seconds describing what constitutes an acceptable data gap. The output of the application provides a packetWriter command line set that includes the exact start and stop times that define the data gap. For each such gap, the data recovery gap detection application will generate a packetWriter command line set necessary for recovering the lost data.

4.10.2 Real-Time Database Gap Detector

This application is written in Labview and is designed to detect gaps as they exist in the various real-time databases. Using the gaps identified with time information relative to the execution of COR dumps, the user can determine if data is being received properly by packetGrabber and real-time databases; there should be no significant data gaps prior to the latest COR dump. The user provides the computer and database to query, the database table name, and the gap size to detect. The gaps detected are reported back in a window on the application and are stored in a user-selected text file.

4.10.3 SDDS Data Mapper

This application is written in Labview and is designed to provide a summary of all the acceleration data for a user provided time period. As a minimum, this application will be executed at the completion of a calendar month to provide an image of data availability for all the sensors that PIMS supports. The user can request this summary information for an individual sensor or for all the sensors. User inputs are the time periods of interest, the sensors to analyze, and a time gap. The output is a text file containing the data gaps greater than the requested time gap and a JPG image illustrating data availability for the requested sensors.

5 Initial Operational Configuration

The purpose of this section is to briefly describe the Flight 6A operational configuration and the utilization of the software components described in section 4. The initial configuration of PIMS computers at the TSC consists of ten computers. Five computers are dedicated to the five SAMS-II sensors on board the ISS. Each such computer is dedicated to receiving, processing, and archiving the data from one SAMS-II sensor. There are three computers dedicated to processing MAMS data; one for MAMS OSS, one for MAMS HiRAP, and one for executing the MOIDP application. The MAMS OSS and MAMS HiRAP computers are dedicated to receiving, processing, and archiving the data from their respective MAMS sensors. The MOIDP computer is responsible for all the data processing and products generated by the MOIDP application (see section 4.9). A single computer is dedicated to the monitor and control application described in section 4.5. The final computer is dedicated to recovering any lost data using the data recovery application.

6 References

- [1] K. McPherson, Software Requirements for Processing Microgravity Acceleration Data from the International Space Station, PIMS-ISS-001, May, 2000
- [2] PIMS Brochure,
http://microgravity.grc.nasa.gov/MSD/MSD_documents/PIMS_Brochure.pdf
- [3] Space Acceleration Measurement System II Data and Command Format Definition Document, SAMS-II-400 Rev. A, December, 1999
- [4] MAMS Software Requirements Document, SP 401010, November, 2000
- [5] Space Station Reference Coordinate Systems, SSP30219
- [6] Space Station Interior and Exterior Operational Location Coding System, SSP30575
- [7] Rice, J.E., OARE Technical Report #152, OARE STS-87 (USMP-4) Final Report, CSI-9801, March, 1998.
- [8] R.V. Hogg, "Adaptive Robust Procedures: A Partial Review and Some Suggestions for Future Applications and Theory," Journal of American Statistical Association, Vol. 69, Number 348, December 1974, pp 909-927.
- [9] T. DeWet, J.W.J. VanWyk, "Efficiency and Robustness of Hogg's Adaptive Trim Means," Commun. Statist.-Theor. Meth., A8(2), pp 117-128, 1979.

Appendix A – Directory Locations for PIMS Software Applications

This appendix contains the directory locations for many of the various software components described in this document.

Software Component	Machine	Directory
Packetgrabber.Conf	Local Host	/etc
Configuration Files For Packetgrabber.Conf	Justice	/home/pims/packetGrabber_configs
Packetgrabber Executable Control	Local Host	/etc/rc.d/init.d
Listener Executable Control	Local Host	/etc/rc.d/init.d
Ancillary Databases	Justice	/var/lib/mysql/pad
Configuration Files For Displays	Justice	/home/pims/launcher_config
Mysql Database Files	Local Host	/var/lib/mysql/pims
Packetwriter Executable	Local Host	/usr/local/bin/pims
Data Gap Detection Executables	Alomar	/home/pims/Gap_check_executables
Database Interface Tool	Justice	Desktop icon
Monitor And Control Executable	Justice	Desktop icon

Appendix B – Ancillary Data Parameters

This appendix describes the relationship between the various ancillary data parameters and the closure of a PAD file in the event a particular ancillary data parameter change is detected. The relationship between ancillary data parameter and file closure are broken down by data type in the tables that follow.

SAMS-II with Ancillary Data Sources

Ancillary Parameter	Ancillary Data Source	File Closure if Parameter Change	Parameter Applicable to this Data Type
T-0	Data Stream	NO	YES
Sampling Rate ⁺	Data Stream	YES	YES
Cutoff Frequency ⁺	Calculated from Data Stream	YES	YES
Gain	Data Stream	NO	YES
Head ID [~]	Data Stream	NO	YES
Location	Ancillary Database	YES	YES
Orientation	Ancillary Database	YES	YES
Bias Coefficients [~]	Ancillary Database	NO	YES
Scale Factor [~]	Ancillary Database	NO	YES
Coordinate System	Ancillary Database	YES	YES
Data Quality Measure	Ancillary Database	YES	YES
Station Configuration	Ancillary Database	YES	YES

+ - These parameters are tied together and one cannot change without changing the other. Specifically, the sampling rate is obtained from the data stream and the cutoff frequency is derived from that sampling rate

~ - Bias and scale factor numbers will only change for a TSH being re-calibrated (on the ground) or when a TSH swap out is required on board. Such a change is handled automatically, since packetGrabber separates data already based on the head ID.

MAMS OSS with Ancillary Data Sources

Ancillary Parameter	Ancillary Data Source	File Closure if Parameter Change	Parameter Applicable to this Data Type
T-0	Data Stream	NO	YES
Sampling Rate ~	Fixed @ 10 samp/sec	NO	YES
Cutoff Frequency ~	Fixed @ 1 Hz	NO	YES
Gain	Not Applicable	-----	NO
Head ID +	Fixed = MAMS OSS	NO	YES
Location	Ancillary Database	YES	YES
Orientation	Ancillary Database	YES	YES
Bias Coefficients	Ancillary Database	YES	YES
Scale Factor &	Ancillary Database	YES	YES
Coordinate System	Ancillary Database	YES	YES
Data Quality Measure	Ancillary Database	YES	YES
Station Configuration	Ancillary Database	YES	YES

~ - The sampling rate and the cutoff frequency are fixed for MAMS OSS Data

+ - For MAMS data in general, the packetGrabber already separates data based on the data type (OSS, HiRAP, or other).

& - There will be a fixed set of scale factor numbers for MAMS OSS. No on board scale factor calibrations will be performed as it was with the OARE.

MAMS HiRAP with Ancillary Data Sources

Ancillary Parameter	Ancillary Data Source	File Closure if Parameter Change	Parameter Applicable to this Data Type
T-0	Data Stream	NO	YES
Sampling Rate ~	Fixed @ 1000 samp/sec	NO	YES
Cutoff Frequency ~	Fixed @ 100 Hz	NO	YES
Gain	Not Applicable	-----	NO
Head ID	Fixed = MAMS HiRAP	NO	YES
Location	Ancillary Database	YES	YES
Orientation	Ancillary Database	YES	YES
Bias Coefficients +	Not Applicable	-----	NO
Scale Factor +	Ancillary Database	NO	YES
Coordinate System	Ancillary Database	YES	YES
Data Quality Measure	Ancillary Database	YES	YES
Station Configuration	Ancillary Database	YES	YES

~ - The sampling rate and the cutoff frequency are fixed for MAMS HiRAP Data

+ - There will be a fixed set of scale factor numbers for MAMS HiRAP. These numbers are a combination of bias and scale factor information. Therefore, no bias coefficients are necessary for MAMS HiRAP provided the scale factor data is properly tracked.

GSE Packet Data with Ancillary Data Sources

Ancillary Parameter	Ancillary Data Source	File Closure if Parameter Change	Parameter Applicable to this Data Type
T-0	Data Stream	NO	YES
Sampling Rate	Data Stream	NO	YES
Cutoff Frequency	Not Applicable	-----	NO
Gain	Not Applicable	-----	NO
Head ID	Not Applicable	-----	NO
Location	Not Applicable	-----	NO
Orientation	Not Applicable	-----	NO
Bias Coefficients	Not Applicable	-----	NO
Scale Factor	Not Applicable	-----	NO
Coordinate System	Ancillary Database	NO	YES
Data Quality Measure	Ancillary Database	YES	YES
Station Configuration	Ancillary Database	YES	YES

The rates and angles data files will contain the ISS body rates and body angles and the ISS center of gravity/mass. This information is stored together for use during MAMS OSS data mapping to alternate locations within the ISS.

Appendix C- SAMS-II and MAMS Data Rates

MAMS APID 903 Data Rates

Packet Type	Packet Rate/Data Rate	Header Size (Bytes)	Packet Size (Bytes)
HiRAP	1 packet/0.192 second	60	1172
Text Messages	Asynchronous	60	68-134
OSS	1 packet/16 seconds	60	1074
Housekeeping	1 packet/15 minutes (default) ¹	60	70
Stored OSS	200 kbps (default) ²	60	1074

1 – Commandable between 0 and 1440 minutes

2 – Commandable between 20 and 200 kbps

SAMS-II APID 890 Data Rates

Sampling Rate	Cutoff Frequency	Packet Rate/Data Rate	Header Size (Bytes)	Packet Size (Bytes)
62.5	25	2 packets/second	60	548 and 564
125	50	2 packets/second	60	1044 and 1060
250	100	4 packets/second	60	2 @ 1236 and 2 @ 868
500	200	8 packets/second	60	6 @ 1236 and 2 @ 500
1000	400	14 packets/second	60	12 @ 1236 and 2 @ 948

Appendix D – Hogg Adaptive Trimean Filter Used In Processing Oare Acceleration Measurements

The following paragraphs were extracted in total from reference [7]. Minor adjustments to the referenced paragraphs and equation numbers were made. These paragraphs provide an excellent summary of the trimmean filter process as it relates to the processing of quasi-steady acceleration data. In this particular case, OARE instrument data is discussed, but the trimmean filter explanation has direct relevance to the MAMS OSS data.

Trimmean Filter Explanation Extracted from Reference [7]:

The OARE instrument is designed to measure the quasi-steady acceleration from below 10 nano-Gs up to 25 milli-Gs and over the quasi-steady bandwidth from 10^{-5} to 10^{-1} Hz. The quasi-steady acceleration components of primary interest are those due to gravity gradient, on-orbit drag, inertial rotations, and perhaps long period venting or gas leaks. However, the instrument is subjected to higher amplitude and higher frequency accelerations (such as structural vibration, station keeping thruster firings, and crew effects) in addition to the quasi-steady accelerations. These higher level accelerations are not well characterized nor statistically invariant over the OARE measurement periods.

In order to obtain a more optimum estimate of the quasi-steady acceleration under the conditions of intermittent thruster firings and crew activities, a robust adaptive estimator has been implemented. For a discussion of robust estimators, see Reference [8]. The particular estimator implemented in OARE is known as the Hogg Adaptive Trimean estimator and is described in more detail in Reference [9].

In essence, the trimmean adaptive filter removes a percentage of the distribution from each tail and then calculates the mean of the remaining distribution. It first measures the departure of the sample distribution from a normal (Gaussian) distribution as measured by a parameter called Q , then adaptively chooses the amount of the trim to be used on the distribution, and finally calculates the mean of the remaining distribution after the trim. This filter is designed to remove the effect of a contaminating distribution (such as a thruster firing) superimposed on a normal distribution (of instrument noise, high frequency vibrations, crew activities, quasi-steady accelerations, etc.),

As implemented, Q is defined by the following equation:

$$Q = [U(20\%) - L(20\%)]/[U(50\%) - L(50\%)] \quad (\text{eq. 1}), \quad \text{where}$$

$U(X\%)$ is the average of the upper $X\%$ of the ordered sample, and
 $L(X\%)$ is the average of the lower $X\%$ of the ordered sample.

In the OARE case, the ordered sample has been a sample of 500 acceleration measurements of the A/D output over a 50-second measurement period.

Q is a measure of the outlier content in the sample. For a Gaussian distribution, Q is 1.75; for samples which have larger tails, $Q > 1.75$. The value of Q is used to estimate the extent that the quasi-steady acceleration measurements may be contaminated by thruster firings, crew activities, etc.

In order to improve the estimate of the quasi-steady acceleration, a trimmean is used to estimate the mean of the quasi-steady population. A trim parameter alpha is determined by the following algorithm:

$$\alpha(Q) = \begin{cases} 0.05 & \text{for } Q \leq 1.75 \\ 0.5 + 0.35 * (Q-1.75)/(2-1.75) & \text{for } 1.75 < Q < 2.0 \text{ (eq. 2)} \\ 0.4 & \text{for } Q \geq 2.0, \end{cases}$$

where alpha is the fraction of the distribution which is trimmed off each tail of the ordered distribution before the mean of the remainder of the distribution is calculated.

Then, for an underlying distribution of n points or measurements with a value of alpha, the trimmean is given by

$$\text{trimmean} = [X_{(k+1)} + X_{(k+2)} + \dots + X_{(n-k)}] / (n-2*k), \text{ (eq. 3),}$$

where $k = \alpha * n$ (eq. 4) and

$X_{(i)}, X_{(i+1)}, \dots, X_{(n)}$, is the ordered set of n points making up the sample distribution.

In summary, OARE measures the quasi-steady level of acceleration for each axis every 25 seconds by taking the trimmean of 50 seconds of A/D counts (500 samples in total) according to equations 1 through 4. It should be noted that for large pulses in one direction, the effect of the trimmean is to shift the estimate of the mean; it does not preserve the DC component in this case.

The trimmean is particularly appropriate for estimating the bias of the OARE instrument, since one wishes to remove the effect of the thruster on the bias measurement if a thruster firing should occur during the bias measurement period.

Appendix Z – Open Work

Date Opened	Problem Description
3/27/01	Consider allowing multiple coordinate systems and multiple mapping locations to be specified. For a MAMS OSS plot, this could allow data to be mapped to the PCS location, but displayed in OSS coordinates and PCS coordinates
8/1/01	Remove references to the final TMF and rewrite with references only to the best available TMF since MAMS bias values are proving to be quite stable